BS EN 14908-6:2014

BSI Standards Publication

Open Data Communication in Building Automation, Controls and Building Management — Control Network Protocol

Part 6: Application elements

bsi.

...making excellence a habit.™

National foreword

This British Standard is the UK implementation of EN 14908-6:2014. It supersedes BS EN 14908-6:2010 which is withdrawn.

The UK participation in its preparation was entrusted to Technical Committee RHE/16, Performance requirements for control systems.

A list of organizations represented on this committee can be obtained on request to its secretary.

This publication does not purport to include all the necessary provisions of a contract. Users are responsible for its correct application.

Published by BSI Standards Limited 2014

ISBN 978 0 580 85153 7

ICS 35.240.99; 91.140.01; 97.120

Compliance with a British Standard cannot confer immunity from legal obligations.

This British Standard was published under the authority of the Standards Policy and Strategy Committee on 31 December 2014.

Amendments/corrigenda issued since publication

Date	Text affected

EUROPEAN STANDARD

NORME EUROPÉENNE

EUROPÄISCHE NORM

EN 14908-6

December 2014

ICS 35.240.99; 91.140.01; 97.120

Supersedes EN 14908-6:2010

English Version

Open Data Communication in Building Automation, Controls and Building Management - Control Network Protocol - Part 6: Application elements

Réseau ouvert de communication de données pour l'automatisation, la régulation et la gestion technique du bâtiment - Protocole de réseau pour le bâtiment - Partie 6 : Eléments pour l'application

Firmenneutrale Datenkommunikation für die Gebäudeautomation und Gebäudemanagement - Gebäude Netzwerk Protokoll - Teil 6: Anwendungselemente

This European Standard was approved by CEN on 18 October 2014.

CEN members are bound to comply with the CEN/CENELEC Internal Regulations which stipulate the conditions for giving this European Standard the status of a national standard without any alteration. Up-to-date lists and bibliographical references concerning such national standards may be obtained on application to the CEN-CENELEC Management Centre or to any CEN member.

This European Standard exists in three official versions (English, French, German). A version in any other language made by translation under the responsibility of a CEN member into its own language and notified to the CEN-CENELEC Management Centre has the same status as the official versions.

CEN members are the national standards bodies of Austria, Belgium, Bulgaria, Croatia, Cyprus, Czech Republic, Denmark, Estonia, Finland, Former Yugoslav Republic of Macedonia, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, Netherlands, Norway, Poland, Portugal, Romania, Slovakia, Slovenia, Spain, Sweden, Switzerland, Turkey and United Kingdom.

EUROPEAN COMMITTEE FOR STANDARDIZATION
COMITÉ EUROPÉEN DE NORMALISATION
EUROPÄISCHES KOMITEE FÜR NORMUNG

CEN-CENELEC Management Centre: Avenue Marnix 17, B-1000 Brussels

Ref. No. EN 14908-6:2014 E

Contents

Page

Foreword

This document (EN 14908-6:2014) has been prepared by Technical Committee CEN/TC 247 “Building automation, controls and building management”, the secretariat of which is held by SNV.

This European Standard shall be given the status of a national standard, either by publication of an identical text or by endorsement, at the latest by June 2015 and conflicting national standards shall be withdrawn at the latest by June 2015.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. CEN [and/or CENELEC] shall not be held responsible for identifying any or all such patent rights.

This document supersedes EN 14908-6:2010.

This European Standard is part of the EN 14908 series, which consists of the following parts:

— EN 14908-1, *Open Data Communication in Building Automation, Controls and Building Management — Control Network Protocol — Part 1: Protocol Stack*

— EN 14908-2, *Open Data Communication in Building Automation, Controls and Building Management — Control Network Protocol — Part 2: Twisted Pair Communication*

— EN 14908-3, *Open Data Communication in Building Automation, Controls and Building Management — Control Network Protocol — Part 3: Power Line Channel Specification*

— EN 14908-4, *Open Data Communication in Building Automation, Controls and Building Management — Control Network Protocol — Part 4: IP Communication*

— EN 14908-5, *Open Data Communication in Building Automation, Controls and Building Management Implementation Guideline — Control Network Protocol — Part 5: Implementation*

— EN 14908-6, *Open Data Communication in Building Automation, Controls and Building Management — Control Network Protocol — Part 6: Application Elements*

The content of this European Standard covers the data communications used for management, automation/control and field functions.

According to the CEN-CENELEC Internal Regulations, the national standards organizations of the following countries are bound to implement this European Standard: Austria, Belgium, Bulgaria, Croatia, Cyprus, Czech Republic, Denmark, Estonia, Finland, Former Yugoslav Republic of Macedonia, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, Netherlands, Norway, Poland, Portugal, Romania, Slovakia, Slovenia, Spain, Sweden, Switzerland, Turkey and the United Kingdom.

Introduction

This European Standard is intended to be used by all involved in design, manufacture, engineering, installation and commissioning activities.

This document specifies the definition of various types of data-transfer containers for carrying temperature, pressure, state, and other forms of data over the protocol defined by EN 14908-1 from any number of devices to any other number of devices, as defined by the limits in EN 14908-1. It also specifies the data types for configuration information, used to define timing, default values, and other data. This document describes a data-file transfer method that may be used for transferring configuration information to and from devices, and specifies the device-interface format that describes the interconnection points of a device.

NOTE The file-transfer mechanism is not exclusively used for loading configuration information into a device and can therefore have other purposes.

This European Standard contributes to the general European policy for energy savings, particularly in the field of the "Energy Performance of Building Directive" and the Construction Products Directive (ER No. 6 "Energy Economy and Heat Retention").

1 Scope

This European Standard provides mechanisms through which various vendors of building automation, control, and building management systems may exchange information in a standardized way.

This document provides specifications for the Application Elements of Control Network Protocol packets as follows:

— definitions of standardized packet (network-variable) data types;

— definitions of device-interface files;

— definitions of standardized configuration-property types;

— definitions of standardized enumeration types;

— definitions of standardized functional profiles;

— definition of the standardized method of file transfer between devices.

The purpose of this specification is to ensure interoperability between various CNP implementations. This document contains all the information necessary to read and interpret the format of data and control information that is used by EN 14908-5. It also defines the device interface for a device as specified, which is necessary to exchange data between various devices from different manufacturers.

2 Normative references

The following documents, in whole or in part, are normatively referenced in this document and are indispensable for its application. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

EN 14908-1:2005, *Open Data Communication in Building Automation, Controls and Building Management — Building Network Protocol — Part 1: Protocol Stack*

EN 14908-2, *Open Data Communication in Building Automation, Controls and Building Management — Control Network Protocol — Part 2: Twisted Pair Communication*

EN 14908-3, *Open Data Communication in Building Automation, Controls and Building Management — Control Network Protocol — Part 3: Power Line Channel Specification*

EN 14908-4, *Open Data Communication in Building Automation, Controls and Building Management — Control Network Protocol — Part 4: IP Communication*

EN 14908-5, *Open Data Communication in Building Automation, Controls and Building Management Implementation Guideline — Control Network Protocol — Part 5: Implementation*

3 Terms and definitions

For the purposes of this document, the terms and definitions given in EN 14908-1:2005 and the following apply.

3.1
application set
functional block or functional blocks to which a configuration property applies

EXAMPLE A network variable, a series or compilation of network variables, a functional block, a series or compilation of functional blocks, or the entire device.

3.2
base type
fundamental type that can be used as the basis of a network-variable type or configuration-property type

Note 1 to entry: The available base types are defined in EN 14908-5.

3.3
changeable-type network variable
network variable whose type can be changed during installation

3.4
CNP device
hardware and software that runs an application and communicates with other devices using the EN 14908-1 protocol

Note 1 to entry: It may optionally interface with input/output hardware. A CNP device includes at least one processor and a CNP transceiver also called a CNP node, or simply a node.

3.5
CNP network
collection of intelligent devices that communicate with each other using the EN 14908-1 protocol over one or more communications channels

3.6
configuration property
CP
data value used to configure the application program in a device

Note 1 to entry: Configuration properties are used to set parameters such as maximum, minimum, default, and override values. CPs are implemented using configuration network variables or as data items within configuration files. Configuration-property data are kept in a device's non-volatile memory.

3.7
configuration-property member
part of a functional profile

Note 1 to entry: See 3.24.

3.8
configuration-property member number
part of a functional profile

Note 1 to entry: See 3.25.

3.9
configuration-property type index
16-bit number that uniquely identifies a configuration-property type within the scope defined by the scope number and program-ID template of the resource file that contains the configuration-property type definition

3.10
device
logical and physical entity of the network containing an application that is designed to communicate with other logical and physical entities

3.11
device channel Identifier
number that optionally specifies the channel to which a device is attached

3.12
device class
two-byte field identifying the primary function of a device and part of the SPID of the device

3.13
device interface
network-visible interface to a device consisting of the unique node ID, program ID, channel ID, location field, device self-documentation string, device configuration properties, and functional blocks

3.14
device-location field
string or number that optionally specifies the location of a device

3.15
device self-documentation string
DSDS
string that specifies the structure of the contents of the self-documentation strings, the functional blocks, and optionally describes the function of a device

3.16
device subclass
two-byte field specifying the usage in the first byte and the channel type in the second byte and is part of the SPID of a device

Note 1 to entry: See the usage and channel-type definitions.

3.17
dynamic functional block
functional block that is added to a device by a network tool after the device is installed

3.18
dynamic network variable
network variable that is added to a device by a network tool after the device is installed

3.19
format
<program ID> four-bit value defining the structure of the program ID as being a Standard Program Identifier (SPID) and device self-documentation string (DSDS) in the device

<resource file> string that provides formatting instructions for a network-variable or configuration-property type

3.20
functional block
fblock
portion of a device's application that performs a task by receiving configuration and operational data inputs, processing the data, and sending operational data outputs

Note 1 to entry: A functional block may receive inputs from the network, from hardware attached to the device, and/or from other functional blocks on a device. A functional block may send outputs to the network, to hardware attached to the device, and/or to other functional blocks on the device. A functional block is an implementation of a functional profile. A "standard" functional block is one based on a standard functional profile template (SFPT).

3.21
functional-block index
sequentially assigned number identifying a functional-block implementation on a device

3.22
functional profile
FP
template that describes common units of functional behaviour, also known as profiles, or FPs; which can be represented with a machine-readable functional-profile template (FPT)

Note 1 to entry: Each functional profile consists of a profile description and a specified set of network variables and configuration properties designed to perform a single function on a device. The network variables and configuration properties specified by the functional profile are called the functional-profile members. A functional profile specifies whether the implementation of each functional-profile member is mandatory or optional. A profile is uniquely identified by a program-ID template, scope, and functional-profile number.

3.23
functional-profile key
functional-profile number

Note 1 to entry: See 3.26.

3.24
functional-profile member
network-variable or configuration-property member of a functional profile

Note 1 to entry: Each functional-profile member is identified as mandatory or optional by the functional profile. Each member also includes a text description of the member for the functional profile.

3.25
functional-profile member number
two-byte number that uniquely identifies a network-variable or configuration-property member of a functional profile

Note 1 to entry: This member number is used to associate a network variable or configuration property on a device with the corresponding network-variable or configuration-property member of the functional profile. Member numbers should be in the range of 1 to 4 095, and need not be contiguous. Member numbers shall be unique, with the exception that network-variable and configuration-property members may use the same number (therefore, network-variable members' numbers should be unique, and configuration-property members' numbers should be unique, but they need not be unique between network-variable members and configuration-property members). There may be a maximum of 255 mandatory members and 255 optional members of each type (scope 0 NV, inheriting NV, scope 0 CP, and inheriting CP).

3.26
functional-profile number
two-byte number that uniquely identifies a functional profile within the scope defined by the scope number and program-ID template of the resource file that contains the functional-profile definition

Note 1 to entry: The functional profile number is also called the "functional-profile key", or the "FPT key".

3.27
functional-profile selector
ASCII vertical bar ("|") or ASCII number sign ("#") to denote the association of a network variable or configuration property with a scope-0 profile or a profile of a higher-numbered scope, respectively, where a higher-numbered scope would denote the NV or CP was added to enhance a scope-0 profile or that it applies to a non-standard profile

EXAMPLE The "|1" member of a functional profile is not the same as the "#1" member of the same profile. This prevents conflicts if new members are added to a standard functional profile that has already been used as the basis for inheriting profiles.

Note 1 to entry: If the functional profile selector is a vertical bar, the member number identifies a member of a scope-0 profile. If the functional profile selector is a number sign, the member number identifies a member of the inheriting profile. The number-sign functional profile selector is always used for members of user functional profiles, including profiles that do not use inheritance. The vertical-bar functional profile selector is always used for members of standard functional

profiles. Two different functional profile members may have the same member number as long as they use different functional profile selectors.

3.28
functional-profile template
functional profile in human- and machine-readable form

Note 1 to entry: See 3.22.

3.29
global index
functional-block index

Note 1 to entry: See 3.21.

3.30
inheriting profile
functional profile that inherits members from a scope-0 profile

3.31
interoperability
conditions that ensure multiple devices − from the same or different manufacturers − can be integrated into a single network without requiring custom device or tool development

3.32
manufacturer Identifier
MID
20-bit number that uniquely identifies the device manufacturer of a device and is part of the device's SPID

3.33
network-interface selection
form of network-variable selection that occurs on the network interface

3.34
network variable
NV
data item that a particular device application program expects to get from other devices on a network (an input network variable) or expects to make available to other devices on a network (an output network variable)

EXAMPLE A temperature, switch value, and actuator-position setting.

Note 1 to entry: Network variable data are typically stored in a device's volatile memory.

3.35
network-variable declaration
establishment of an instance of a network variable type within the code of an application

3.36
network-variable index
sequentially assigned number identifying a network variable implementation on a device

Note 1 to entry: For Neuron C applications, the index is assigned by the Neuron C compiler in the order of declaration. The first network variable on a device has index 0, the second index 1, etc.

3.37
network-variable member
functional-profile member that is a network variable

Note 1 to entry: See 3.24.

3.38
network-variable member number
number of a functional-profile member that is a network variable

Note 1 to entry: See 3.25.

3.39
network-variable programmatic name
name assigned to a network-variable implementation by the device application developer

Note 1 to entry: The programmatic name is limited to 16 characters, including any optional prefixes. The programmatic name is not significant for interoperability, but conventions are suggested in EN 14908-5 to make programmatic names easier to use for integrators.

3.40
network-variable selection
process of associating a network-variable selector with a network variable on a device

3.41
network-variable type
specification of the length, units, valid range, and resolution of the data contained within a network variable

Note 1 to entry: A network variable type may be a simple, one, two, or four-byte scalar type; or a more complex structure or union of up to 31 bytes.

3.42
network-variable type index
16-bit number that uniquely identifies a network-variable type within the scope defined by the scope number and program-ID template of the resource file that contains the network-variable type definition

3.43
unique node Identifier
UID
unique 48-bit identifier within the read-only data structure of a device as defined by the EN 14908-1 protocol

Note 1 to entry: It is also called the "unique_node_ID".

3.44
node
<common> device

<precise> physical and logical presence on a CNP network with a unique node ID and network address

Note 1 to entry: The unique node ID relates to the identification of a single instance of an implemented EN 14908-1 protocol stack. A device is also a network presence with an application processor and one or more nodes. A device with multiple unique node IDs would consist of multiple nodes. Some infrastructure devices, such as routers, also consist of more than one unique node ID and thus consist of multiple nodes.

3.45
passive configuration tool
PCT
network tool that can be used on a device to assist in the successful commissioning of the device without disrupting the operation of other network tools

Note 1 to entry: It may be a plug-in, standalone software, hardware attachment, or other tool. A passive configuration tool has attributes and capabilities as defined in EN 14908-5.

3.46
primary functional block
functional block on a device that implements the most important function for the device

3.47
primary functional profile
functional profile that defines the primary functional block on a device

3.48
proprietary data
data and message definitions in the device interface that are known only to the manufacturer and the manufacturer's agents

3.49
self-documentation string
SD string
text string associated with a device, network variable, or configuration property that is stored within a device and within the device interface (XIF) file for a device

Note 1 to entry: Network tools can read the self-documentation strings from the device itself or from the device interface file.

3.50
self-documentation text
optional text within a device, network variable, or configuration property self-documentation string that provides documentation of the intended use of the device, network variable, or configuration property respectively for use by integrators

3.51
shared-media channel
communications channel where messages can leak between tools and devices belonging to different systems

3.52
SNVT_xxx
placeholder for SNVT chosen by the manufacturer − at the time of the device's development − for a network variable that is defined by a functional profile, but where the type for the network variable is not defined by the profile

Note 1 to entry: See definition of SNVT in 4.1.

3.53
standard configuration-property type
SCPT
configuration-property type that has been standardized by this document in Clause 4

Note 1 to entry: A SCPT is a standardized definition of the units, scaling, encoding, valid range, and meaning of the contents of configuration properties.

3.54
standard network-variable type
SNVT
network-variable type that has been standardized by this document in 4.1

3.55
standard program Identifier
SPID
eight-byte number that uniquely identifies the device interface for a device

Note 1 to entry: Encoded according to rules specified in Annex B.

3.56
static functional block
functional block that is statically defined for a device; that is, a functional block that is not a dynamic functional block

3.57
static network variable
network variable that is statically defined for a device; that is, a network variable that is not a dynamic network variable

3.58
subsystem
two or more devices working together to perform a function and bearing fixed, pre-defined relationships to one another

Note 1 to entry: A subsystem may use one or more EN 14908-1 domains.

3.59
successful commissioning
<noun> process of taking a device and integrating it into a network

<adjective> device can be physically installed in a network and made to perform its application function with the exclusive use of its device interface and a choice of third-party tools

3.60
system
one or more independently managed subsystems working together to perform a function

Note 1 to entry: A system may use one or more EN 14908-1 domains.

3.61
unconfigured device
device without a valid network configuration

3.62
usage
one-byte value describing the intended usage of the device and is part of the SPID of a device

Note 1 to entry: The usage field consists of a one-bit changeable-interface flag, a one-bit functional-profile-specific flag, and a 6-bit usage ID.

3.63
usage Identifier
six-bit value in the least-significant portion of the usage field that identifies the primary intended usage of a device

3.64
user data
non-standardized user functional blocks, user network variables, and user configuration properties used by a device manufacturer to augment the device interface

3.65
wink function
function provided by a device that allows a network integrator to physically identify the device

EXAMPLE A wink function may blink an LED on the device.

4 Standard network-variable type – SNVT

4.1 Introduction

CNP devices typically exchange data using network variables. Network variables greatly simplify the tasks of designing CNP application programs for interoperability with multiple vendors' products. A network variable is any data item (temperature, a switch value, or an actuator position setting) that a particular device application program expects to get from other devices on the network (an *input network variable*) or expects to make available to other devices on the network (an *output network variable*). When the application program has a changed value for an output network variable, it simply passes the new value to the device firmware. Via a process that takes place during network design and installation called *binding*, the protocol stack is configured to know the logical address of the other devices or group of devices in the network expecting that network variable, and it assembles and sends the appropriate packets to these devices. Similarly, when the protocol stack receives an updated value for an input network variable required by its application program, it passes the data to the application program. The binding process thus creates logical *connections* between an output network variable in one device and an input network variable in another device or group of devices. Connections may be thought of as "virtual wires". Every network variable has a type that defines the units, scaling, and structure of the data contained within the network variable. Network variables shall be the same type to be connected. This prevents common installation errors from occurring such as a pressure output being connected to a temperature input. Type translators are available to convert network variables of one type to another type. This document defines standard network-variable types (SNVTs) that define commonly used types. Alternatively, manufacturers may define their own user network-variable types (UNVTs). UNVTs shall be documented in manufacturer-specific resource files.

Basic data types (e.g. signed long) and their sizes are defined in EN 14908-5.

4.2 SNVT_amp

Overview: Electric current in Amperes.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
1	2	signed long	-32 768	32 767	$1* 10^{-1} * (Raw+0)$	0,1 Amperes

4.3 SNVT_amp_mil

Overview: Electric current in milliAmperes.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
2	2	signed long	-32 768	32 767	$1* 10^{-1} * (Raw+0)$	0,1 milliAmperes

4.4 SNVT_angle

Overview: Angular distance in radians.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
3	2	unsigned long	0	65 535	$1* 10^{-3} * (Raw+0)$	0,001 radians

4.5 SNVT_angle_vel

Overview: Angular velocity in radians/second.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
4	2	signed long	-32 768	32 767	$1* 10^{-1} * (Raw+0)$	0,1 radians/second

4.6 SNVT_btu_kilo

Overview: Thermal energy in kilo-Btus.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
5	2	unsigned long	0	65 535	$1* 10^{0} * (Raw+0)$	1 kilo-Btus

4.7 SNVT_btu_mega

Overview: Thermal energy in mega-Btus.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
6	2	unsigned long	0	65 535	$1* 10^{0} * (Raw+0)$	1 mega-Btus

4.8 SNVT_char_ascii

Overview: ASCII character in 8-bit ASCII character.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
7	1	unsigned char	0	255	$1* 10^{0} * (Raw+0)$	1 8-bit ASCII character

4.9 SNVT_count

Overview: Absolute count in units.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
8	2	unsigned long	0	65 535	$1* 10^{0} * (Raw+0)$	1 units

4.10 SNVT_count_inc

Overview: Increment count in units (delta).

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
9	2	signed long	-32 768	32 767	$1* 10^{0} * (Raw+0)$	1 units (delta)

4.11 SNVT_date_day

Overview: Day of week.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
11	1	days_of_week_t	-1	6	none (enumeration)	-1

4.12 SNVT_elec_kwh

Overview: Electrical energy in kiloWatt-hours.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
13	2	unsigned long	0	65 535	$1* 10^{0} * (Raw+0)$	1 kW-h

4.13 SNVT_elec_whr

Overview: Electrical energy in Watt-hours.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
14	2	unsigned long	0	65 535	$1* 10^{-1} * (Raw+0)$	0,1 W-h

4.14 SNVT_flow

Overview: Flow volume in liters/second.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
15	2	unsigned long	0	65 535	$1* 10^{0} * (Raw+0)$	1 l/s

4.15 SNVT_flow_mil

Overview: Flow volume in milliliters/second.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
16	2	unsigned long	0	65 535	$1* 10^{0} * (Raw+0)$	1 ml/s

4.16 SNVT_length

Overview: Length in meters.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
17	2	unsigned long	0	65 535	$1* 10^{-1} * (Raw+0)$	0,1 m

4.17 SNVT_length_kilo

Overview: Length in kilometers.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
18	2	unsigned long	0	65 535	$1* 10^{-1} * (Raw+0)$	0,1 km

4.18 SNVT_length_micr

Overview: Length in micrometers (microns).

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
19	2	unsigned long	0	65 535	$1* 10^{-1} * (Raw+0)$	0,1 µm (microns)

4.19 SNVT_length_mil

Overview: Length in millimeters.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
20	2	unsigned long	0	65 535	$1* 10^{-1} * (Raw+0)$	0,1 mm

4.20 SNVT_lev_cont

Overview: Continuous level in % of full level.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
21	1	unsigned short	0	200	$5* 10^{-1} * (Raw+0)$	0,5 % of full level

4.21 SNVT_mass

Overview: Mass in grams.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
23	2	unsigned long	0	65 535	$1* 10^{-1} * (Raw+0)$	0,1 g

4.22 SNVT_mass_kilo

Overview: Mass in kilograms.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
24	2	unsigned long	0	65 535	$1* 10^{-1} * (Raw+0)$	0,1 kg

4.23 SNVT_mass_mega

Overview: Mass in metric tons.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
25	2	unsigned long	0	65 535	$1* 10^{-1} * (Raw+0)$	0,1 t

4.24 SNVT_mass_mil

Overview: Mass in milligrams.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
26	2	unsigned long	0	65 535	$1* 10^{-1} * (Raw+0)$	0,1 mg

4.25 SNVT_power

Overview: Power in Watts.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
27	2	unsigned long	0	65 535	$1* 10^{-1} * (Raw+0)$	0,1 W

4.26 SNVT_power_kilo

Overview: Power in kiloWatts.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
28	2	unsigned long	0	65 535	$1* 10^{-1} * (Raw+0)$	0,1 kW

4.27 SNVT_ppm

Overview: Concentration in ppm.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
29	2	unsigned long	0	65 535	$1* 10^{0} * (Raw+0)$	1 ppm

4.28 SNVT_press

Overview: Pressure (gauge) in kiloPascals.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
30	2	signed long	-32 768	32 767	$1* 10^{-1} * (Raw+0)$	0,1 kP

4.29 SNVT_res

Overview: Electric resistance in Ohms.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
31	2	unsigned long	0	65 535	$1* 10^{-1} * (Raw+0)$	0,1 Ohms

4.30 SNVT_res_kilo

Overview: Electric resistance in kiloOhms.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
32	2	unsigned long	0	65 535	$1* 10^{-1} * (Raw+0)$	0,1 kiloOhms

4.31 SNVT_sound_db

Overview: Sound level in dB.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
33	2	signed long	-32 768	32 767	$1* 10^{-2} * (Raw+0)$	0,01 dB

4.32 SNVT_speed

Overview: Linear velocity in meters/second.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
34	2	unsigned long	0	65 535	$1* 10^{-1} * (Raw+0)$	0,1 m/s

4.33 SNVT_speed_mil

Overview: Linear velocity in meters/second.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
35	2	unsigned long	0	65 535	$1* 10^{-3} * (Raw+0)$	0,001 m/s

4.34 SNVT_str_asc

Overview: Character string (30 characters max).

Index	Size in Bytes	CNP Data Type			
36	31	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_str_asc.ascii			unsigned char	255	$1* 10^{0} * (Raw+0)$
			(none)	0	1 array of 31 characters

4.35 SNVT_str_int

Overview: Wide character string with locale code (15 characters max).

Index	Size in Bytes	CNP Data Type			
37	31	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_str_int.char_set			unsigned short	255	$1* 10^{0} * (Raw+0)$
			(none)	0	1 code value
SNVT_str_int.wide_char			unsigned long	65 535	$1* 10^{0} * (Raw+0)$
			(none)	0	1 array of 15 wide characters

4.36 SNVT_telcom

Overview: Telecomm states.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
38	1	telcom_states_t	-1	20	none (enumeration)	-1

4.37 SNVT_temp

Overview: Temperature in degrees Celsius.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
39	2	unsigned long	0	65 535	$1* 10^{-1} * (Raw-2740)$	0,1 °C

4.38 SNVT_vol

Overview: Volume in liters.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
41	2	unsigned long	0	65 535	$1* 10^{-1} * (Raw+0)$	0,1 l

4.39 SNVT_vol_kilo

Overview: Volume in kiloliters.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
42	2	unsigned long	0	65 535	$1* 10^{-1} * (Raw+0)$	0,1 kl

4.40 SNVT_vol_mil

Overview: Volume in milliliters.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
43	2	unsigned long	0	65 535	$1* 10^{-1} * (Raw+0)$	0,1 ml

4.41 SNVT_volt

Overview: Electric potential in Volts.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
44	2	signed long	-32 768	32 767	$1* 10^{-1} * (Raw+0)$	0,1 V

4.42 SNVT_volt_dbmv

Overview: Electric potential in dB microVolts.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
45	2	signed long	-32 768	32 767	$1* 10^{-2} * (Raw+0)$	0,01 dB microVolts

4.43 SNVT_volt_kilo

Overview: Electric potential in kiloVolts.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
46	2	signed long	-32 768	32 767	$1* 10^{-1} * (Raw+0)$	0,1 kV

4.44 SNVT_volt_mil

Overview: Electric potential in milliVolts.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
47	2	signed long	-32 768	32 767	$1* 10^{-1} * (Raw+0)$	0,1 mV

4.45 SNVT_amp_f

Overview: Electric current in Amperes.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
48	4	floating-point	-3,402 823 466 385 3E + 038	3,402 823 466 385 3E + 038	floating	NAN and Out of Range

4.46 SNVT_angle_f

Overview: Angular distance in radians.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
49	4	floating-point	-3,402 823 466 385 3E + 038	3,402 823 466 385 3E + 038	floating	NAN and Out of Range

4.47 SNVT_angle_vel_f

Overview: Angular velocity in radians/second.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
50	4	floating-point	-3,402 823 466 385 3E + 038	3,402 823 466 385 3E + 038	floating	NAN and Out of Range

4.48 SNVT_count_f

Overview: Absolute count in units.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
51	4	floating-point	0	3,402 823 466 385 3E + 038	floating	NAN and Out of Range

4.49 SNVT_count_inc_f

Overview: Increment count in units (delta).

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
52	4	floating-point	-3,402 823 466 385 3E + 038	3,402 823 466 385 3E + 038	floating	NAN and Out of Range

4.50 SNVT_flow_f

Overview: Flow volume in liters/second.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
53	4	floating-point	-3,402 823 466 385 3E + 038	3,402 823 466 385 3E + 038	floating	NAN and Out of Range

4.51 SNVT_length_f

Overview: Length in meters.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
54	4	floating-point	0	3,402 823 466 385 3E + 038	floating	NAN and Out of Range

4.52 SNVT_lev_cont_f

Overview: Continuous level in % of full scale.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
55	4	floating-point	0	100	floating	NAN and Out of Range

4.53 SNVT_mass_f

Overview: Mass in grams.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
56	4	floating-point	0	3,402 823 466 385 3E + 038	floating	NAN and Out of Range

4.54 SNVT_power_f

Overview: Power in Watts.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
57	4	floating-point	-3,402 823 466 385 3E + 038	3,402 823 466 385 3E + 038	floating	NAN and Out of Range

4.55 SNVT_ppm_f

Overview: Concentration in ppm.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
58	4	floating-point	0	3,402 823 466 385 3E + 038	floating	NAN and Out of Range

4.56 SNVT_press_f

Overview: Pressure (gauge) in Pascals.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
59	4	floating-point	-3,402 823 466 385 3E + 038	3,402 823 466 385 3E + 038	floating	NAN and Out of Range

4.57 SNVT_res_f

Overview: Electric resistance in Ohms.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
60	4	floating-point	0	3,402 8234663853E+038	floating	NAN and Out of Range

4.58 SNVT_sound_db_f

Overview: Sound level in dBspl.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
61	4	floating-point	-3,402 823 466 385 3E + 038	3,402 823 466 385 3E + 038	floating	NAN and Out of Range

4.59 SNVT_speed_f

Overview: Linear velocity in meters/second.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
62	4	floating-point	-3,402 823 466 385 3E + 038	3,402 823 466 385 3E + 038	floating	NAN and Out of Range

4.60 SNVT_temp_f

Overview: Temperature in degrees Celsius.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
63	4	floating-point	-273,170 013 427 73	3,402 823 466 385 3E + 038	floating	NAN and Out of Range

4.61 SNVT_time_f

Overview: Elapsed time in seconds.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
64	4	floating-point	0	3,402 823 466 385 3E + 038	floating	NAN and Out of Range

4.62 SNVT_vol_f

Overview: Volume in liters.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
65	4	floating-point	0	3,402 823 466 385 3E + 038	floating	NAN and Out of Range

4.63 SNVT_volt_f

Overview: Electric potential in Volts.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
66	4	floating-point	-3,402 823 466 385 3E + 038	3,402 823 466 385 3E + 038	floating	NAN and Out of Range

4.64 SNVT_btu_f

Overview: Thermal energy in Btus.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
67	4	floating-point	0	3,402 823 466 385 3E + 038	floating	NAN and Out of Range

4.65 SNVT_elec_whr_f

Overview: Electrical energy in Watt-hours.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
68	4	floating-point	0	3,402 823 466 385 3E + 038	floating	NAN and Out of Range

4.66 SNVT_config_src

Overview: Network configuration source.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
69	1	config_source_t	-1	1	none (enumeration)	-1

4.67 SNVT_color

Overview: CIELAB color.

Index	Size in Bytes	CNP Data Type			
70	6	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_color.L_star			unsigned long	1 000	$1* 10^{-1}$ * (Raw+0)
			(none)	0	0,1
SNVT_color.a_star			signed long	2 000	$1* 10^{-1}$ * (Raw+0)
			(none)	-2 000	0,1
SNVT_color.b_star			signed long	2 000	$1* 10^{-1}$ * (Raw+0)
			(none)	-2 000	0,1

4.68 SNVT_grammage

Overview: Density in grams/sq meter.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
71	2	unsigned long	0	65 535	$1* 10^{-1}$ * (Raw+0)	0,1 g/m^2

4.69 SNVT_grammage_f

Overview: Density in grams/sq meter.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
72	4	floating-point	0	3,402 823 466 385 3E + 038	floating	NAN and Out of Range

4.70 SNVT_file_req

Overview: File request.

Index	Size in Bytes	CNP Data Type			
73	12	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution

SNVT_file_req.request	file_request_t	6	none (enumeration)
	-1	-1	1
SNVT_file_req.index	unsigned long	65 535	$1* 10^0 * (Raw+0)$
	(none)	0	1 file index
SNVT_file_req.receive_timeout	unsigned long	65 535	$1* 10^0 * (Raw+0)$
	(none)	0	1 ms
SNVT_file_req.dest_address	union		
SNVT_file_req.dest_address.addrt	structure		
	EN 14908-1 address in device's internal address table entry		
SNVT_file_req.dest_address.addrt.type	unsigned short	33	$1* 10^0 * (Raw+0)$
	(none)	33	1 8-bit unsigned value
SNVT_file_req.dest_address.addrt.index	unsigned long	65535	$1* 10^0 * (Raw+0)$
	(none)	0	1 16-bit unsigned value
SNVT_file_req.dest_address.sn	structure		
SNVT_file_req.dest_address.sn.type	unsigned short	1	$1* 10^0 * (Raw+0)$
	(none)	1	1 8-bit unsigned value
SNVT_file_req.dest_address.sn.domain	bitfield	1	$1* 10^0 * (Raw+0)$
		0	1 (bits: 1) (offset: 0)
SNVT_file_req.dest_address.sn.node	bitfield	127	$1* 10^0 * (Raw+0)$
		0	1 (bits: 7) (offset: 1)
SNVT_file_req.dest_address.sn.retry	bitfield	15	$1* 10^0 * (Raw+0)$
		0	1 (bits: 4) (offset: 4)
SNVT_file_req.dest_address.sn.tx_timer	bitfield	15	$1* 10^0 * (Raw+0)$
		0	1 (bits: 4) (offset: 4)
SNVT_file_req.dest_address.sn.subnet	unsigned short	255	$1* 10^0 * (Raw+0)$

	0	0	1 subnet number
SNVT_file_req.dest_address.gp	structure		
SNVT_file_req.dest_address.gp.type	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 0)
SNVT_file_req.dest_address.gp.size	bitfield	65	1* 10^0 * (Raw+0)
		0	1 (bits: 7) (offset: 1)
SNVT_file_req.dest_address.gp.domain	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 0)
SNVT_file_req.dest_address.gp.unused	bitfield	0	1* 10^0 * (Raw+0)
		0	1 (bits: 7) (offset: 1)
SNVT_file_req.dest_address.gp.retry	bitfield	15	1* 10^0 * (Raw+0)
		0	1 (bits: 4) (offset: 4)
SNVT_file_req.dest_address.gp.tx_timer	bitfield	15	1* 10^0 * (Raw+0)
		0	1 (bits: 4) (offset: 4)
SNVT_file_req.dest_address.gp.group	unsigned short	255	1* 10^0 * (Raw+0)
	(none)	0	1
SNVT_file_req.auth_on	unsigned short	1	1* 10^0 * (Raw+0)
	(none)	0	1 boolean
SNVT_file_req.prio_on	unsigned short	1	1* 10^0 * (Raw+0)
	(none)	0	1 boolean

4.71 SNVT_file_status

Overview: File status.

Index	Size in Bytes	CNP Data Type			
74	27	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution

SNVT_file_status.status	file_status_t	11	none (enumeration)
	-1	-1	1
SNVT_file_status.number_of_files	unsigned long	65 535	1* 10^0 * (Raw+0)
	(none)	0	1 count
SNVT_file_status.selected_file	unsigned long	65 535	1* 10^0 * (Raw+0)
	(none)	0	1 file index
SNVT_file_status.adr	union		
SNVT_file_status.adr.descriptor	structure		
SNVT_file_status.adr.descriptor.file_info	signed char	127	1* 10^0 * (Raw+0)
	(none)	-128	1 array of 16 characters
SNVT_file_status.adr.descriptor.size	s32_type (signed 32-bit type)	2147483647	1* 10^0 * (Raw+0)
	(none)	0	1 bytes
SNVT_file_status.adr.descriptor.type	unsigned long	65 535	1* 10^0 * (Raw+0)
	(none)	0	1
SNVT_file_status.adr.address	structure		
SNVT_file_status.adr.address.domain_id	unsigned short	255	1* 10^0 * (Raw+0)
	(none)	0	1 array of 6 bytes
SNVT_file_status.adr.address.domain_length	unsigned short	6	1* 10^0 * (Raw+0)
	(none)	0	1
SNVT_file_status.adr.address.subnet	unsigned short	255	1* 10^0 * (Raw+0)
	0	0	1 subnet number
SNVT_file_status.adr.address.node	unsigned short	127	1* 10^0 * (Raw+0)
	0	0	1 node number

4.72 SNVT_freq_f

Overview: Frequency in Hertz.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
75	4	floating-point	0	3,402 823 466 385 3E + 038	floating	NAN and Out of Range

4.73 SNVT_freq_hz

Overview: Frequency in Hertz.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
76	2	unsigned long	0	65 535	$1* 10^{-1} * (Raw+0)$	0,1 Hz

4.74 SNVT_freq_kilohz

Overview: Frequency in kiloHertz.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
77	2	unsigned long	0	65 535	$1* 10^{-1} * (Raw+0)$	0,1 kHz

4.75 SNVT_freq_milhz

Overview: Frequency in miliHertz.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
78	2	unsigned long	0	65 535	$1* 10^{-4} * (Raw+0)$	0,000 1 Hz

4.76 SNVT_lux

Overview: Illumination in lux.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
79	2	unsigned long	0	65 535	$1* 10^{0} * (Raw+0)$	1 lx

4.77 SNVT_lev_percent

Overview: Percentage level in % of full level. Level percent. SNVT_switch should be used instead of SNVT_lev_percent, with the exception of network variables that are used to communicate a percentage value and that require the additional resolution provided by SNVT_lev_percent; or for network variable members of functional profiles that are designed primarily for interfacing with SNVT_lev_percent members of other profiles.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
81	2	signed long	-32 768	32 767	$5* 10^{-3} * (Raw+0)$	0,005 % of full level

4.78 SNVT_multiplier

Overview: Multiplier.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
82	2	unsigned long	0	65 535	$5* 10^{-4} * (Raw+0)$	0,000 5 16-bit unsigned value

4.79 SNVT_state

Overview: State vector. Each state is a boolean single bit value. SNVT_state_64 is preferred.

Index	Size in Bytes	CNP Data Type			
83	2	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_state.bit0			bitfield	1	$1* 10^{0} * (Raw+0)$
				0	1 (bits: 1) (offset: 0)
SNVT_state.bit1			bitfield	1	$1* 10^{0} * (Raw+0)$
				0	1 (bits: 1) (offset: 1)
SNVT_state.bit2			bitfield	1	$1* 10^{0} * (Raw+0)$
				0	1 (bits: 1) (offset: 2)
SNVT_state.bit3			bitfield	1	$1* 10^{0} * (Raw+0)$
				0	1 (bits: 1) (offset: 3)
SNVT_state.bit4			bitfield	1	$1* 10^{0} * (Raw+0)$
				0	1 (bits: 1) (offset: 4)
SNVT_state.bit5			bitfield	1	$1* 10^{0} * (Raw+0)$
				0	1 (bits: 1) (offset: 5)
SNVT_state.bit6			bitfield	1	$1* 10^{0} * (Raw+0)$
				0	1 (bits: 1) (offset: 6)
SNVT_state.bit7			bitfield	1	$1* 10^{0} * (Raw+0)$
				0	1 (bits: 1) (offset: 7)
SNVT_state.bit8			bitfield	1	$1* 10^{0} * (Raw+0)$

		0	1 (bits: 1) (offset: 0)
SNVT_state.bit9	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 1)
SNVT_state.bit10	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 2)
SNVT_state.bit11	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 3)
SNVT_state.bit12	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 4)
SNVT_state.bit13	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 5)
SNVT_state.bit14	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 6)
SNVT_state.bit15	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 7)

4.80 SNVT_time_stamp

Overview: Time stamp.

Index	**Size in Bytes**	**CNP Data Type**			
84	7	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_time_stamp.year			signed long	3 000	1* 10^0 * (Raw+0)
			-1	-1	1 year
SNVT_time_stamp.month			unsigned short	12	1* 10^0 * (Raw+0)
			(none)	0	1 month
SNVT_time_stamp.day			unsigned short	31	1* 10^0 * (Raw+0)
			(none)	0	1 day

SNVT_time_stamp.hour	unsigned short	23	$1* 10^{0} * (Raw+0)$
	(none)	0	1 h
SNVT_time_stamp.minute	unsigned short	59	$1* 10^{0} * (Raw+0)$
	(none)	0	1 min
SNVT_time_stamp.second	unsigned short	59	$1* 10^{0} * (Raw+0)$
	(none)	0	1 s

4.81 SNVT_zerospan

Overview: Zero and span. Linear transformation parameters: multiply by the span-factor, then add the zero-term.

Index	Size in Bytes	CNP Data Type			
85	4	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_zerospan.zero			signed long	32 767	$5* 10^{-3} * (Raw+0)$
			(none)	-32 768	0,005 16-bit signed value
SNVT_zerospan.span			unsigned long	65 535	$5* 10^{-4} * (Raw+0)$
			(none)	0	0,000 5 16-bit unsigned value

4.82 SNVT_magcard

Overview: ISO 7811 magnetic card stripe data.

Index	Size in Bytes	CNP Data Type			
86	20	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_magcard.digit1			bitfield	11	$1* 10^{0} * (Raw+0)$
				11	1 (bits: 4) (offset: 0)
SNVT_magcard.digit2			bitfield	15	$1* 10^{0} * (Raw+0)$
				0	1 (bits: 4) (offset: 4)

SNVT_magcard.digit3	bitfield	15	1* 10^0 * (Raw+0)
		0	1 (bits: 4) (offset: 0)
SNVT_magcard.digit4	bitfield	15	1* 10^0 * (Raw+0)
		0	1 (bits: 4) (offset: 4)
SNVT_magcard.digit5	bitfield	15	1* 10^0 * (Raw+0)
		0	1 (bits: 4) (offset: 0)
SNVT_magcard.digit6	bitfield	15	1* 10^0 * (Raw+0)
		0	1 (bits: 4) (offset: 4)
SNVT_magcard.digit7	bitfield	15	1* 10^0 * (Raw+0)
		0	1 (bits: 4) (offset: 0)
SNVT_magcard.digit8	bitfield	15	1* 10^0 * (Raw+0)
		0	1 (bits: 4) (offset: 4)
SNVT_magcard.digit9	bitfield	15	1* 10^0 * (Raw+0)
		0	1 (bits: 4) (offset: 0)
SNVT_magcard.digit10	bitfield	15	1* 10^0 * (Raw+0)
		0	1 (bits: 4) (offset: 4)
SNVT_magcard.digit11	bitfield	15	1* 10^0 * (Raw+0)
		0	1 (bits: 4) (offset: 0)
SNVT_magcard.digit12	bitfield	15	1* 10^0 * (Raw+0)
		0	1 (bits: 4) (offset: 4)
SNVT_magcard.digit13	bitfield	15	1* 10^0 * (Raw+0)
		0	1 (bits: 4) (offset: 0)
SNVT_magcard.digit14	bitfield	15	1* 10^0 * (Raw+0)
		0	1 (bits: 4) (offset: 4)
SNVT_magcard.digit15	bitfield	15	1* 10^0 * (Raw+0)
		0	1 (bits: 4) (offset: 0)
SNVT_magcard.digit16	bitfield	15	1* 10^0 * (Raw+0)

		0	1 (bits: 4) (offset: 4)
SNVT_magcard.digit17	bitfield	15	$1* 10^{0} *$ (Raw+0)
		0	1 (bits: 4) (offset: 0)
SNVT_magcard.digit18	bitfield	15	$1* 10^{0} *$ (Raw+0)
		0	1 (bits: 4) (offset: 4)
SNVT_magcard.digit19	bitfield	15	$1* 10^{0} *$ (Raw+0)
		0	1 (bits: 4) (offset: 0)
SNVT_magcard.digit20	bitfield	15	$1* 10^{0} *$ (Raw+0)
		0	1 (bits: 4) (offset: 4)
SNVT_magcard.digit21	bitfield	15	$1* 10^{0} *$ (Raw+0)
		0	1 (bits: 4) (offset: 0)
SNVT_magcard.digit22	bitfield	15	$1* 10^{0} *$ (Raw+0)
		0	1 (bits: 4) (offset: 4)
SNVT_magcard.digit23	bitfield	15	$1* 10^{0} *$ (Raw+0)
		0	1 (bits: 4) (offset: 0)
SNVT_magcard.digit24	bitfield	15	$1* 10^{0} *$ (Raw+0)
		0	1 (bits: 4) (offset: 4)
SNVT_magcard.digit25	bitfield	15	$1* 10^{0} *$ (Raw+0)
		0	1 (bits: 4) (offset: 0)
SNVT_magcard.digit26	bitfield	15	$1* 10^{0} *$ (Raw+0)
		0	1 (bits: 4) (offset: 4)
SNVT_magcard.digit27	bitfield	15	$1* 10^{0} *$ (Raw+0)
		0	1 (bits: 4) (offset: 0)
SNVT_magcard.digit28	bitfield	15	$1* 10^{0} *$ (Raw+0)
		0	1 (bits: 4) (offset: 4)
SNVT_magcard.digit29	bitfield	15	$1* 10^{0} *$ (Raw+0)
		0	1 (bits: 4) (offset: 0)

SNVT_magcard.digit30	bitfield	15	$1* 10^0 * (Raw+0)$
		0	1 (bits: 4) (offset: 4)
SNVT_magcard.digit31	bitfield	15	$1* 10^0 * (Raw+0)$
		0	1 (bits: 4) (offset: 0)
SNVT_magcard.digit32	bitfield	15	$1* 10^0 * (Raw+0)$
		0	1 (bits: 4) (offset: 4)
SNVT_magcard.digit33	bitfield	15	$1* 10^0 * (Raw+0)$
		0	1 (bits: 4) (offset: 0)
SNVT_magcard.digit34	bitfield	15	$1* 10^0 * (Raw+0)$
		0	1 (bits: 4) (offset: 4)
SNVT_magcard.digit35	bitfield	15	$1* 10^0 * (Raw+0)$
		0	1 (bits: 4) (offset: 0)
SNVT_magcard.digit36	bitfield	15	$1* 10^0 * (Raw+0)$
		0	1 (bits: 4) (offset: 4)
SNVT_magcard.digit37	bitfield	15	$1* 10^0 * (Raw+0)$
		0	1 (bits: 4) (offset: 0)
SNVT_magcard.digit38	bitfield	15	$1* 10^0 * (Raw+0)$
		0	1 (bits: 4) (offset: 4)
SNVT_magcard.digit39	bitfield	15	$1* 10^0 * (Raw+0)$
		0	1 (bits: 4) (offset: 0)
SNVT_magcard.digit40	bitfield	15	$1* 10^0 * (Raw+0)$
		0	1 (bits: 4) (offset: 4)

4.83 SNVT_elapsed_tm

Overview: Elapsed time.

Index	**Size in Bytes**	**CNP Data Type**			
87	7	structure			
Field			CNP Data Type /	Maximum /	Scaled Value /

	Invalid Value	Minimum	Resolution
SNVT_elapsed_tm.day	unsigned long	65 535	1* 10^0 * (Raw+0)
	65 535	0	1 day
SNVT_elapsed_tm.hour	unsigned short	23	1* 10^0 * (Raw+0)
	(none)	0	1 h
SNVT_elapsed_tm.minute	unsigned short	59	1* 10^0 * (Raw+0)
	(none)	0	1 min
SNVT_elapsed_tm.second	unsigned short	59	1* 10^0 * (Raw+0)
	(none)	0	1 s
SNVT_elapsed_tm.millisecond	unsigned long	999	1* 10^0 * (Raw+0)
	(none)	0	1 ms

4.84 SNVT_alarm

Overview: Alarm status.

Index	**Size in Bytes**	**CNP Data Type**			
88	29	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_alarm.location			unsigned short	255	1* 10^0 * (Raw+0)
			(none)	0	1 array of 6 bytes
SNVT_alarm.object_id			unsigned long	65 535	1* 10^0 * (Raw+0)
			(none)	0	1 object index
SNVT_alarm.alarm_type			alarm_type_t	32	none (enumeration)
			-1	-13	1
SNVT_alarm.priority_level			priority_level_t	11	none (enumeration)
			-1	-1	1
SNVT_alarm.index_to_SNVT			unsigned long	65 535	1* 10^0 * (Raw+0)
			(none)	0	1 index of NV causing alarm

SNVT_alarm.value	unsigned short	255	$1* 10^0 * (Raw+0)$
	(none)	0	1 array of 4 bytes
SNVT_alarm.year	unsigned long	3 000	$1* 10^0 * (Raw+0)$
	(none)	0	1 year
SNVT_alarm.month	unsigned short	12	$1* 10^0 * (Raw+0)$
	(none)	0	1 month
SNVT_alarm.day	unsigned short	31	$1* 10^0 * (Raw+0)$
	(none)	0	1 day
SNVT_alarm.hour	unsigned short	23	$1* 10^0 * (Raw+0)$
	(none)	0	1 h
SNVT_alarm.minute	unsigned short	59	$1* 10^0 * (Raw+0)$
	(none)	0	1 min
SNVT_alarm.second	unsigned short	59	$1* 10^0 * (Raw+0)$
	(none)	0	1 s
SNVT_alarm.millisecond	unsigned long	999	$1* 10^0 * (Raw+0)$
	(none)	0	1 ms
SNVT_alarm.alarm_limit	unsigned short	255	$1* 10^0 * (Raw+0)$
	(none)	0	1 array of 4 bytes

4.85 SNVT_currency

Overview: Currency.

Index	**Size in Bytes**	**CNP Data Type**	
89	6	structure	

Field	CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_currency.currency	currency_t	56	none (enumeration)
	-1	-1	1
SNVT_currency.power_of_10	signed short	127	$1* 10^0 * (Raw+0)$

	(none)	-128	1 power of 10
SNVT_currency.value	s32_type (signed 32-bit type)	214 748 364 7	1* 10^0 * (Raw+0)
	(none)	-214 748 364 8	1 currency value

4.86 SNVT_file_pos

Overview: File position.

Index	Size in Bytes	CNP Data Type			
90	6	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_file_pos.rw_ptr			s32_type (signed 32-bit type)	2 147 483 647	1* 10^0 * (Raw+0)
			(none)	0	1 file byte address
SNVT_file_pos.rw_length			unsigned long	65 535	1* 10^0 * (Raw+0)
			(none)	0	1 number of bytes

4.87 SNVT_muldiv

Overview: Multiply/Divide gain factor

Index	Size in Bytes	CNP Data Type			
91	4	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_muldiv.multiplier			unsigned long	65 535	1* 10^0 * (Raw+0)
			(none)	0	1 16-bit unsigned value
SNVT_muldiv.divisor			unsigned long	65 535	1* 10^0 * (Raw+0)
			0	1	1 16-bit unsigned value

4.88 SNVT_obj_request

Overview: Object request.

Index	Size in Bytes	CNP Data Type	
92	3	structure	

Field	CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_obj_request.object_id	unsigned long	65 535	1* 10^0 * (Raw+0)
	(none)	0	1 object index
SNVT_obj_request.object_request	object_request_t	24	none (enumeration)
	-1	-1	1

4.89 SNVT_obj_status

Overview: Object status.

Index	Size in Bytes	CNP Data Type	
93	6	structure	

Field	CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_obj_status.object_id	unsigned long	65 535	1* 10^0 * (Raw+0)
	(none)	0	1 object index
SNVT_obj_status.invalid_id	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 0)
SNVT_obj_status.invalid_request	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 1)
SNVT_obj_status.disabled	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 2)
SNVT_obj_status.out_of_limits	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 3)
SNVT_obj_status.open_circuit	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 4)
SNVT_obj_status.out_of_service	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 5)
SNVT_obj_status.mechanical_fault	bitfield	1	1* 10^0 * (Raw+0)

		0	1 (bits: 1) (offset: 6)
SNVT_obj_status.feedback_failure	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 7)
SNVT_obj_status.over_range	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 0)
SNVT_obj_status.under_range	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 1)
SNVT_obj_status.electrical_fault	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 2)
SNVT_obj_status.unable_to_measure	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 3)
SNVT_obj_status.comm_failure	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 4)
SNVT_obj_status.fail_self_test	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 5)
SNVT_obj_status.self_test_in_progress	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 6)
SNVT_obj_status.locked_out	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 7)
SNVT_obj_status.manual_control	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 0)
SNVT_obj_status.in_alarm	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 1)
SNVT_obj_status.in_override	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 2)
SNVT_obj_status.report_mask	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 3)

SNVT_obj_status.programming_mode	bitfield	1	$1* 10^0$ * (Raw+0)
		0	1 (bits: 1) (offset: 4)
SNVT_obj_status.programming_fail	bitfield	1	$1* 10^0$ * (Raw+0)
		0	1 (bits: 1) (offset: 5)
SNVT_obj_status.alarm_notify_disabled	bitfield	1	$1* 10^0$ * (Raw+0)
		0	1 (bits: 1) (offset: 6)
SNVT_obj_status.reset_complete	bitfield	1	$1* 10^0$ * (Raw+0)
		0	1 (bits: 1) (offset: 7)
SNVT_obj_status.reserved2	bitfield	0	$1* 10^0$ * (Raw+0)
		0	1 (bits: 8) (offset: 0)

4.90 SNVT_preset

Overview: Preset.

Index	**Size in Bytes**	**CNP Data Type**			
94	14	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_preset.learn			learn_mode_t	3	none (enumeration)
			-1	-1	1
SNVT_preset.selector			unsigned long	65 535	$1* 10^0$ * (Raw+0)
			(none)	0	1 16-bit unsigned value
SNVT_preset.value			unsigned short	255	$1* 10^0$ * (Raw+0)
			(none)	0	1 array of 4 bytes
SNVT_preset.day			unsigned long	65 535	$1* 10^0$ * (Raw+0)
			65 535	0	1 day
SNVT_preset.hour			unsigned short	23	$1* 10^0$ * (Raw+0)
			(none)	0	1 h
SNVT_preset.minute			unsigned short	59	$1* 10^0$ * (Raw+0)

	(none)	0	1 min
SNVT_preset.second	unsigned short	59	1* 10^0 * (Raw+0)
	(none)	0	1 s
SNVT_preset.millisecond	unsigned long	999	1* 10^0 * (Raw+0)
	(none)	0	1 ms

4.91 SNVT_switch

Overview: Switch.

Index	Size in Bytes	CNP Data Type			
95	2	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_switch.value			unsigned short	200	5* 10^{-1} * (Raw+0)
			(none)	0	0,5 % of full level
SNVT_switch.state			signed short	1	1* 10^0 * (Raw+0)
			-1	-1	1 state code

4.92 SNVT_trans_table

Overview: Translation table.

Index	Size in Bytes	CNP Data Type			
96	30	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_trans_table.point			floating-point	3,402 823 466 385 3E + 038	floating
			NAN and Out of Range	-3,402 823 466 385 3E + 038	Points
SNVT_trans_table.interp_pts_0_to_1			bitfield	1	1* 10^0 * (Raw+0)
				0	1 (bits: 2) (offset: 0)
SNVT_trans_table.interp_pts_1_to_2			bitfield	1	1* 10^0 * (Raw+0)

		0	1 (bits: 2) (offset: 2)
SNVT_trans_table.interp_pts_2_to_3	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 2) (offset: 4)
SNVT_trans_table.interp_pts_3_to_4	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 2) (offset: 6)
SNVT_trans_table.interp_pts_4_to_5	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 2) (offset: 0)
SNVT_trans_table.interp_pts_5_to_6	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 2) (offset: 2)
SNVT_trans_table.interp_pts_6_to_0	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 2) (offset: 4)

4.93 SNVT_override

Overview: Override code.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
97	1	override_t	-1	2	none (enumeration)	-1

4.94 SNVT_pwr_fact

Overview: Power factor.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
98	2	signed long	-20 000	20 000	5* 10^{-5} * (Raw+0)	5E-005 multiplier

4.95 SNVT_pwr_fact_f

Overview: Power factor.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
99	4	floating-point	-1	1	floating	NAN and Out of Range

4.96 SNVT_density

Overview: Density in kilograms/cubic meter.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
100	2	unsigned long	0	65 535	5* 10^{-1} * (Raw+0)	0,5 kg/m^3

4.97 SNVT_density_f

Overview: Density in kilograms/cubic meter.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
101	4	floating-point	0	3,402 823 466 385 3E + 038	floating	NAN and Out of Range

4.98 SNVT_rpm

Overview: Angular velocity in revolutions/minute (RPM).

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
102	2	unsigned long	0	65 535	$1* 10^{0} * (Raw+0)$	1 RPM

4.99 SNVT_hvac_emerg

Overview: HVAC emergency mode.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
103	1	emerg_t	-1	5	none (enumeration)	-1

4.100 SNVT_angle_deg

Overview: Angular distance in degrees.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
104	2	signed long	-17 999	18 000	$2* 10^{-2} * (Raw+0)$	0,02°

4.101 SNVT_temp_p

Overview: Temperature in degrees Celsius.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
105	2	signed long	-27 317	32 767	$1* 10^{-2} * (Raw+0)$	0,01 °C

4.102 SNVT_temp_setpt

Overview: Temperature.

Index	Size in Bytes	CNP Data Type		
106	12	structure		
Field		CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution

SNVT_temp_setpt.occupied_cool	signed long	32 767	$1* 10^{-2} *$ (Raw+0)
	32 767	-27 317	0,01 °C
SNVT_temp_setpt.standby_cool	signed long	32 767	$1* 10^{-2} *$ (Raw+0)
	32 767	-27 317	0,01 °C
SNVT_temp_setpt.unoccupied_cool	signed long	32 767	$1* 10^{-2} *$ (Raw+0)
	32 767	-27 317	0,01 °C
SNVT_temp_setpt.occupied_heat	signed long	32 767	$1* 10^{-2} *$ (Raw+0)
	32 767	-27 317	0,01 °C
SNVT_temp_setpt.standby_heat	signed long	32 767	$1* 10^{-2} *$ (Raw+0)
	32 767	-27 317	0,01 °C
SNVT_temp_setpt.unoccupied_heat	signed long	32 767	$1* 10^{-2} *$ (Raw+0)
	32 767	-27 317	0,01 °C

4.103 SNVT_time_sec

Overview: Elapsed time in seconds.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
107	2	unsigned long	0	65 535	$1* 10^{-1} *$ (Raw+0)	0,1 s

4.104 SNVT_hvac_mode

Overview: HVAC mode.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
108	1	hvac_t	-1	20	none (enumeration)	-1

4.105 SNVT_occupancy

Overview: Occupancy.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
109	1	occup_t	-1	3	none (enumeration)	-1

4.106 SNVT_area

Overview: Area in square meters.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution

110	2	unsigned long	0	65 535	2* 10^{-4} * (Raw+0)	0,000 2 m^2

4.107 SNVT_hvac_overid

Overview: HVAC override.

Index	Size in Bytes	CNP Data Type			
111	5	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_hvac_overid.state			hvac_overid_t	48	none (enumeration)
			-1	-1	1
SNVT_hvac_overid.percent			signed long	32 767	5* 10^{-3} * (Raw+0)
			32 767	-32 768	0,005 % of full scale
SNVT_hvac_overid.flow			unsigned long	65 535	1* 10^{0} * (Raw+0)
			65 535	0	1 l/s

4.108 SNVT_hvac_status

Overview: HVAC status.

Index	Size in Bytes	CNP Data Type			
112	12	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_hvac_status.mode			hvac_t	17	none (enumeration)
			-1	-1	1
SNVT_hvac_status.heat_output_primary			signed long	32 767	5* 10^{-3} * (Raw+0)
			32 767	-32 768	0,005 % of full scale
SNVT_hvac_status.heat_output_secondary			signed long	32 767	5* 10^{-3} * (Raw+0)
			32 767	-32 768	0,005 % of full scale
SNVT_hvac_status.cool_output			signed long	32 767	5* 10^{-3} * (Raw+0)
			32 767	-32 768	0,005 % of full scale

SNVT_hvac_status.econ_output	signed long	32 767	$5* 10^{-3} * (Raw+0)$
	32 767	-32 768	0,005 % of full scale
SNVT_hvac_status.fan_output	signed long	32 767	$5* 10^{-3} * (Raw+0)$
	32 767	-32 768	0,005 % of full scale
SNVT_hvac_status.in_alarm	unsigned short	255	$1* 10^{0} * (Raw+0)$
	(none)	0	1 alarm value

4.109 SNVT_press_p

Overview: Pressure (gauge) in Pascals.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
113	2	signed long	-32 768	32 767	$1* 10^{0} * (Raw+0)$	1 Pa

4.110 SNVT_address

Overview: Neuron address as a 16-bit address value.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
114	2	unsigned long	16 384	64 767	$1* 10^{0} * (Raw+0)$	1 16-bit address value

4.111 SNVT_scene

Overview: Scene control.

Index	Size in Bytes	CNP Data Type		
115	2	structure		
Field		CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_scene.function		scene_t	23	none (enumeration)
		-1	-1	1
SNVT_scene.scene_number		unsigned short	255	$1* 10^{0} * (Raw+0)$
		(none)	1	1

4.112 SNVT_scene_cfg

Overview: Scene configuration.

Index	Size in Bytes	CNP Data Type			
116	10	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_scene_cfg.function			scene_config_t	4	none (enumeration)
			-1	-1	1
SNVT_scene_cfg.scene_number			unsigned short	255	$1* 10^{0} * (Raw+0)$
			(none)	1	1
SNVT_scene_cfg.setting			unsigned short	200	$5* 10^{-1} * (Raw+0)$
			255	0	0,5 % of full level
SNVT_scene_cfg.rotation			signed long	18 000	$2* 10^{-2} * (Raw+0)$
			32 767	-17 999	0,02°
SNVT_scene_cfg.fade_time			unsigned long	65 535	$1* 10^{-1} * (Raw+0)$
			65 535	0	0,1 s
SNVT_scene_cfg.delay_time			unsigned long	65 535	$1* 10^{-1} * (Raw+0)$
			65 535	0	0,1 s
SNVT_scene_cfg.scene_priority			unsigned short	255	$1* 10^{0} * (Raw+0)$
			(none)	0	1 priority value

4.113 SNVT_setting

Overview: Setting control.

Index	Size in Bytes	CNP Data Type			
117	4	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_setting.function			setting_t	5	none (enumeration)

	-1	-1	1
SNVT_setting.setting	unsigned short	200	$5* 10^{-1}$ * (Raw+0)
	255	0	0,5 % of full level
SNVT_setting.rotation	signed long	18 000	$2* 10^{-2}$ * (Raw+0)
	32767	-17 999	0,02°

4.114 SNVT_evap_state

Overview: Evaporator state.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
118	1	evap_t	-1	2	none (enumeration)	-1

4.115 SNVT_therm_mode

Overview: Thermostat mode.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
119	1	therm_mode_t	-1	2	none (enumeration)	-1

4.116 SNVT_defr_mode

Overview: Defrost mode.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
120	1	defrost_mode_t	-1	2	none (enumeration)	-1

4.117 SNVT_defr_term

Overview: Defrost termination.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
121	1	defrost_term_t	-1	100	none (enumeration)	-1

4.118 SNVT_defr_state

Overview: Defrost state.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
122	1	defrost_state_t	-1	4	none (enumeration)	-1

4.119 SNVT_time_min

Overview: Elapsed time in minutes.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
123	2	unsigned long	0	65 535	$1* 10^{0} * (Raw+0)$	1 min

4.120 SNVT_time_hour

Overview: Elapsed time in hours.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
124	2	unsigned long	0	65 535	$1* 10^{0} * (Raw+0)$	1 h

4.121 SNVT_ph

Overview: Acidity in pH. Ratio of concentration of ions.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
125	2	signed long	-32 768	32 767	$1* 10^{-3} * (Raw+0)$	0,001 pH

4.122 SNVT_ph_f

Overview: Acidity in pH. Ratio of concentration of ions.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
126	4	floating-point	-3,402 823 466 385 3E + 038	3,402 823 466 385 3E + 038	floating	NAN and Out of Range

4.123 SNVT_chlr_status

Overview: Chiller status.

Index	Size in Bytes	CNP Data Type			
127	3	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_chlr_status.chlr_run_mode			chiller_t	4	none (enumeration)
			-1	-1	1
SNVT_chlr_status.chlr_op_mode			hvac_t	17	none (enumeration)

	-1	-1	1
SNVT_chlr_status.chlr_state	structure		
SNVT_chlr_status.chlr_state.in_alarm	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 0)
SNVT_chlr_status.chlr_state.run_enabled	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 1)
SNVT_chlr_status.chlr_state.local	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 2)
SNVT_chlr_status.chlr_state.limited	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 3)
SNVT_chlr_status.chlr_state.chw_flow	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 4)
SNVT_chlr_status.chlr_state.condw_flow	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 5)

4.124 SNVT_tod_event

Overview: Time-of-day event. Occupancy scheduling event.

Index	**Size in Bytes**	**CNP Data Type**		
128	4	structure		
Field		CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_tod_event.current_state		occup_t	3	none (enumeration)
		-1	-1	1
SNVT_tod_event.next_state		occup_t	3	none (enumeration)
		-1	-1	1
SNVT_tod_event.time_to_next_state		unsigned long	65 535	1* 10^0 * (Raw+0)
		(none)	0	1 min

4.125 SNVT_smo_obscur

Overview: Smoke obscuration in percent obscuration.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
129	2	unsigned long	0	5 000	$1* 10^{-3} * (Raw+0)$	0,001 % obscuration

4.126 SNVT_fire_test

Overview: Fire test request.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
130	1	fire_test_t	-1	3	none (enumeration)	-1

4.127 SNVT_temp_ror

Overview: Temperature rate of change/rise in degrees Celsius/minute.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
131	2	signed long	-32 768	32 767	$5* 10^{-1} * (Raw+0)$	0,5 °C/min

4.128 SNVT_fire_init

Overview: Fire initiator type.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
132	1	fire_initiator_t	-1	16	none (enumeration)	-1

4.129 SNVT_fire_indcte

Overview: Fire indicator type.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
133	1	fire_indicator_t	-1	8	none (enumeration)	-1

4.130 SNVT_time_zone

Overview: Time zone descriptor.

Index	Size in Bytes	CNP Data Type			
134	15	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_time_zone.second_time_offset			s32_type (signed 32-bit type)	86 400	$1* 10^0$ * (Raw+0)
			(none)	-86 400	1 s
SNVT_time_zone.type_of_description			calendar_type_t	2	none (enumeration)
			-1	-1	1
SNVT_time_zone.hour_of_start_DST			unsigned short	23	$1* 10^0$ * (Raw+0)
			(none)	0	1 h
SNVT_time_zone.minute_of_start_DST			unsigned short	59	$1* 10^0$ * (Raw+0)
			(none)	0	1 min
SNVT_time_zone.second_of_start_DST			unsigned short	59	$1* 10^0$ * (Raw+0)
			(none)	0	1 s
SNVT_time_zone.start_DST			union		
			Daylight savings time start day		
SNVT_time_zone.start_DST.G_day_of_start_DST			unsigned long	365	$1* 10^0$ * (Raw+0)
			(none)	0	1 day
SNVT_time_zone.start_DST.J_day_of_start_DST			unsigned long	365	$1* 10^0$ * (Raw+0)
			(none)	1	1 day

SNVT_time_zone.start_DST.M_start_DST	structure		
SNVT_time_zone.start_DST.M_start_DST.month_of_start_DST	bitfield	12	$1* 10^0$ * (Raw+0)
		1	1 (bits: 4) (offset: 0)
SNVT_time_zone.start_DST.M_start_DST.week_of_start_DST	bitfield	5	$1* 10^0$ * (Raw+0)
		1	1 (bits: 3) (offset: 4)
SNVT_time_zone.start_DST.M_start_DST.dateday_of_start_DST	days_of_week_t	6	none (enumeration)
	-1	-1	1
SNVT_time_zone.hour_of_end_DST	unsigned short	23	$1* 10^0$ * (Raw+0)
	(none)	0	1 h
SNVT_time_zone.minute_of_end_DST	unsigned short	59	$1* 10^0$ * (Raw+0)
	(none)	0	1 min
SNVT_time_zone.second_of_end_DST	unsigned short	59	$1* 10^0$ * (Raw+0)
	(none)	0	1 s
SNVT_time_zone.end_DST	union		
	Daylight savings time end day		
SNVT_time_zone.end_DST.G_day_of_end_DST	unsigned long	365	$1* 10^0$ * (Raw+0)
	(none)	0	1 day
SNVT_time_zone.end_DST.J_day_of_end_DST	unsigned long	365	$1* 10^0$ * (Raw+0)
	(none)	1	1 day
SNVT_time_zone.end_DST.M_end_DST	structure		

SNVT_time_zone.end_DST.M_end_DST.month_of_end_DST	bitfield	12	$1* 10^0$ * (Raw+0)
		1	1 (bits: 4) (offset: 0)
SNVT_time_zone.end_DST.M_end_DST.week_of_end_DST	bitfield	5	$1* 10^0$ * (Raw+0)
		1	1 (bits: 3) (offset: 4)
SNVT_time_zone.end_DST.M_end_DST.dateday_of_end_DST	days_of_week_t	6	none (enumeration)
	-1	-1	1

4.131 SNVT_earth_pos

Overview: Earth position.

Index	**Size in Bytes**	**CNP Data Type**			
135	11	structure			
Field		CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution	
SNVT_earth_pos.latitude_direction		bitfield	1	$1* 10^0$ * (Raw+0)	
			0	1 (bits: 1) (offset: 0)	
SNVT_earth_pos.longitude_direction		bitfield	1	$1* 10^0$ * (Raw+0)	
			0	1 (bits: 1) (offset: 1)	
SNVT_earth_pos.latitude_deg		unsigned short	90	$1* 10^0$ * (Raw+0)	
		255	0	1°	
SNVT_earth_pos.latitude_min		unsigned long	59 999	$1* 10^{-3}$ * (Raw+0)	
		65 535	0	0,001 min	
SNVT_earth_pos.longitude_deg		unsigned short	180	$1* 10^0$ * (Raw+0)	
		255	0	1°	
SNVT_earth_pos.longitude_min		unsigned long	59 999	$1* 10^{-3}$ * (Raw+0)	
		65 535	0	0,001 min	

SNVT_earth_pos.height_above_sea	floating-point	3,4028234663853E+038	floating
	NAN and Out of Range	-3,4028234663853E+038	Height above sea level

4.132 SNVT_reg_val

Overview: Register value.

Index	Size in Bytes	CNP Data Type			
136	6	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_reg_val.raw			s32_type (signed 32-bit type)	2 147 483 647	1* 10^0 * (Raw+0)
			(none)	-2 147 483 648	1
SNVT_reg_val.unit			reg_val_unit_t	43	none (enumeration)
			-1	-1	1
SNVT_reg_val.nr_decimals			bitfield	7	1* 10^0 * (Raw+0)
				0	1 (bits: 3) (offset: 0)

4.133 SNVT_reg_val_ts

Overview: Register value.

Index	Size in Bytes	CNP Data Type			
137	13	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_reg_val_ts.raw			s32_type (signed 32-bit type)	2 147 483 647	1* 10^0 * (Raw+0)
			(none)	-2 147 483 648	1
SNVT_reg_val_ts.unit			reg_val_unit_t	43	none (enumeration)
			-1	-1	1
SNVT_reg_val_ts.nr_decimals			bitfield	7	1* 10^0 * (Raw+0)
				0	1 (bits: 3) (offset: 0)

SNVT_reg_val_ts.status	bitfield	15	$1* 10^0 * (Raw+0)$
		0	1 (bits: 4) (offset: 3)
SNVT_reg_val_ts.reg_state	bitfield	1	$1* 10^0 * (Raw+0)$
		0	1 (bits: 1) (offset: 7)
SNVT_reg_val_ts.year	signed long	3 000	$1* 10^0 * (Raw+0)$
	-1	-1	1 year
SNVT_reg_val_ts.month	unsigned short	12	$1* 10^0 * (Raw+0)$
	(none)	0	1 months
SNVT_reg_val_ts.day	unsigned short	31	$1* 10^0 * (Raw+0)$
	(none)	0	1 day
SNVT_reg_val_ts.hour	unsigned short	23	$1* 10^0 * (Raw+0)$
	(none)	0	1 h
SNVT_reg_val_ts.minute	unsigned short	59	$1* 10^0 * (Raw+0)$
	(none)	0	1 min
SNVT_reg_val_ts.second	unsigned short	59	$1* 10^0 * (Raw+0)$
	(none)	0	1 s

4.134 SNVT_volt_ac

Overview: Alternating current electric potential in volts AC.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
138	2	unsigned long	0	65 535	$1* 10^0 * (Raw+0)$	1 volts AC

4.135 SNVT_amp_ac

Overview: Alternating current electric potential in amperes AC.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
139	2	unsigned long	0	65 535	$1* 10^0 * (Raw+0)$	1 amperes AC

4.136 SNVT_turbidity

Overview: Turbidity in nephelometric turbidity units.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
143	2	unsigned long	0	65 535	$1* 10^{-3} * (Raw+0)$	0,001 NTU

4.137 SNVT_turbidity_f

Overview: Turbidity in nephelometric turbidity units.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
144	4	floating-point	0	3,402 823 466 385 3E + 038	floating	NAN and Out of Range

4.138 SNVT_hvac_type

Overview: HVAC unit type.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
145	1	hvac_hvt_t	-1	9	none (enumeration)	-1

4.139 SNVT_elec_kwh_l

Overview: Electrical energy in kiloWatt-hours.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
146	4	s32_type (signed 32-bit type)	-214 748 364 8	214 748 364 7	$1* 10^{-1} * (Raw+0)$	0,1 kW-h

4.140 SNVT_temp_diff_p

Overview: Temp difference in degrees Celsius.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
147	2	signed long	-32 768	32 767	$1* 10^{-2} * (Raw+0)$	0,01 °C

4.141 SNVT_ctrl_req

Overview: Control request.

Index	Size in Bytes	CNP Data Type			
148	5	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_ctrl_req.receiver_id			unsigned long	65 535	1* 10^0 * (Raw+0)
			0	1	1 ID number
SNVT_ctrl_req.sender_id			unsigned long	65 535	1* 10^0 * (Raw+0)
			65 535	1	1 ID number
SNVT_ctrl_req.sender_prio			unsigned short	200	1* 10^0 * (Raw+0)
			(none)	0	1 priority value

4.142 SNVT_ctrl_resp

Overview: Control response.

Index	Size in Bytes	CNP Data Type			
149	7	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_ctrl_resp.status			control_resp_t	5	none (enumeration)
			-1	-1	1
SNVT_ctrl_resp.sender			union		
SNVT_ctrl_resp.sender.id			unsigned long	65 535	1* 10^0 * (Raw+0)
			65 535	1	1 ID number
SNVT_ctrl_resp.sender.range			structure		
SNVT_ctrl_resp.sender.range.lower			unsigned long	65 535	1* 10^0 * (Raw+0)

	65 535	1	1 ID number
SNVT_ctrl_resp.sender.range.upper	unsigned long	65 535	1* 10^{0} * (Raw+0)
	65 535	1	1 ID number
SNVT_ctrl_resp.controller_id	unsigned long	65 535	1* 10^{0} * (Raw+0)
	65 535	1	1 ID number

4.143 SNVT_ptz

Overview: Camera pan-tilt-zoom (PTZ).

Index	**Size in Bytes**	**CNP Data Type**		
150	6	structure		
Field		CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_ptz.pan_dir		pan_dir_t	2	none (enumeration)
		-1	-1	1
SNVT_ptz.pan_speed		unsigned short	250	4* 10^{-1} * (Raw+0)
		(none)	0	0,4 % of full level
SNVT_ptz.tilt_dir		tilt_dir_t	2	none (enumeration)
		-1	-1	1
SNVT_ptz.tilt_speed		unsigned short	250	4* 10^{-1} * (Raw+0)
		(none)	0	0,4 % of full level
SNVT_ptz.zoom		zoom_t	2	none (enumeration)
		-1	-1	1
SNVT_ptz.zoom_speed		unsigned short	250	4* 10^{-1} * (Raw+0)
		(none)	0	0,4 % of full level

4.144 SNVT_privacyzone

Overview: Privacy zone.

Index	Size in Bytes	CNP Data Type			
151	4	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_privacyzone.action			privacyzone_t	5	none (enumeration)
			-1	-1	1
SNVT_privacyzone.number			unsigned short	255	1* 10^0 * (Raw+0)
			0	0	1 zone number
SNVT_privacyzone.camera_id			unsigned long	65 535	1* 10^0 * (Raw+0)
			0	0	1 ID number

4.145 SNVT_pos_ctrl

Overview: Position control.

Index	Size in Bytes	CNP Data Type			
152	13	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_pos_ctrl.receiver_id			unsigned long	65 535	1* 10^0 * (Raw+0)
			0	0	1 ID number
SNVT_pos_ctrl.controller_id			unsigned long	65 535	1* 10^0 * (Raw+0)
			0	0	1 ID number
SNVT_pos_ctrl.controller_prio			unsigned short	100	1* 10^0 * (Raw+0)
			(none)	0	1 priority value
SNVT_pos_ctrl.function			cam_func_t	2	none (enumeration)
			-1	-1	1
SNVT_pos_ctrl.action			cam_act_t	2	none (enumeration)

	-1	-1	1
SNVT_pos_ctrl.value	union		
SNVT_pos_ctrl.value.number	unsigned short	255	$1* 10^{0} * (Raw+0)$
	0	0	1 action number
SNVT_pos_ctrl.value.abspos	structure		
SNVT_pos_ctrl.value.abspos.pan	SNVT_angle_deg		
SNVT_pos_ctrl.value.abspos.tilt	SNVT_angle_deg		
SNVT_pos_ctrl.value.abspos.zoom	SNVT_lev_percent		

4.146 SNVT_enthalpy

Overview: Enthalpy in kiloJoules/kg.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
153	2	signed long	-32 768	32 767	$1* 10^{-2} * (Raw+0)$	0,01 kJ/kg

4.147 SNVT_gfci_status

Overview: GFCI status type.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
154	1	gfci_status_t	-1	5	none (enumeration)	-1

4.148 SNVT_motor_state

Overview: Motor state.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
155	1	motor_state_t	-1	7	none (enumeration)	-1

4.149 SNVT_pumpset_mn

Overview: Pumpset.

Index	**Size in Bytes**	**CNP Data Type**			
156	8	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_pumpset_mn.main_pump			motor_state_t	7	none (enumeration)
			-1	-1	1
SNVT_pumpset_mn.booster_pump			motor_state_t	7	none (enumeration)
			-1	-1	1
SNVT_pumpset_mn.priority_level			priority_level_t	11	none (enumeration)
			-1	-1	1
SNVT_pumpset_mn.process_ready			boolean_t	1	none (enumeration)
			-1	-1	1
SNVT_pumpset_mn.emergency_stop_activated			boolean_t	1	none (enumeration)
			-1	-1	1
SNVT_pumpset_mn.main_pump_drive_enabled			boolean_t	1	none (enumeration)
			-1	-1	1
SNVT_pumpset_mn.booster_pump_drive_enabled			boolean_t	1	none (enumeration)
			-1	-1	1
SNVT_pumpset_mn.maintenance_required			boolean_t	1	none (enumeration)
			-1	-1	1

4.150 SNVT_ex_control

Overview: Exclusive control.

Index	Size in Bytes	CNP Data Type			
157	10	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_ex_control.control_status			ex_control_t	2	none (enumeration)
			-1	-1	1
SNVT_ex_control.control_device_addr			structure		
SNVT_ex_control.control_device_addr.domain_id			unsigned short	255	$1* 10^0 * (Raw+0)$
			(none)	0	1 array of 6 bytes
SNVT_ex_control.control_device_addr.domain_length			unsigned short	6	$1* 10^0 * (Raw+0)$
			(none)	0	1
SNVT_ex_control.control_device_addr.subnet			unsigned short	255	$1* 10^0 * (Raw+0)$
			0	0	1 subnet number
SNVT_ex_control.control_device_addr.node			unsigned short	127	$1* 10^0 * (Raw+0)$
			0	0	1 node number

4.151 SNVT_pumpset_sn

Overview: Pumpset sensor.

Index	Size in Bytes	CNP Data Type			
158	23	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_pumpset_sn.total_dilution_flow			SNVT_flow_mil		

SNVT_pumpset_sn.exhaust_temperature	SNVT_temp		
SNVT_pumpset_sn.exhaust_pressure	SNVT_press		
SNVT_pumpset_sn.shaft_seal_purge_pressure	SNVT_press		
SNVT_pumpset_sn.inlet_vacuum	SNVT_press_f		
SNVT_pumpset_sn.supply_voltage	SNVT_volt		
SNVT_pumpset_sn.coolant_flow	SNVT_flow_mil		
SNVT_pumpset_sn.coolant_flow_low	boolean_t	1	none (enumeration)
	-1	-1	1
SNVT_pumpset_sn.dilution_active	boolean_t	1	none (enumeration)
	-1	-1	1
SNVT_pumpset_sn.ballast_dilution_active	boolean_t	1	none (enumeration)
	-1	-1	1
SNVT_pumpset_sn.inlet_purge_dilution_active	boolean_t	1	none (enumeration)
	-1	-1	1
SNVT_pumpset_sn.exhaust_dilution_active	boolean_t	1	none (enumeration)
	-1	-1	1
SNVT_pumpset_sn.dilution_flow_out_of_range	boolean_t	1	none (enumeration)
	-1	-1	1
SNVT_pumpset_sn.power_supply_on	boolean_t	1	none (enumeration)
	-1	-1	1

4.152 SNVT_pump_sensor

Overview: Pump sensor.

Index	Size in Bytes	CNP Data Type			
159	19	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_pump_sensor.rotational_speed			SNVT_freq_hz		
SNVT_pump_sensor.body_temperature			SNVT_temp		
SNVT_pump_sensor.motor_external_temperature			SNVT_temp		
SNVT_pump_sensor.motor_internal_temperature			SNVT_temp		
SNVT_pump_sensor.motor_overloaded			boolean_t	1	none (enumeration)
			-1	-1	1
SNVT_pump_sensor.oil_level_low			boolean_t	1	none (enumeration)
			-1	-1	1
SNVT_pump_sensor.phase_imbalance_detected			boolean_t	1	none (enumeration)
			-1	-1	1
SNVT_pump_sensor.current_usage			SNVT_amp		
SNVT_pump_sensor.power_usage			SNVT_power_kilo		
SNVT_pump_sensor.temperature_control			unit_temp_t	3	none (enumeration)
			-1	-1	1
SNVT_pump_sensor.electromagnetic_brake_active			boolean_t	1	none (enumeration)

	-1	-1	1
SNVT_pump_sensor.friction_brake_active	boolean_t	1	none (enumeration)
	-1	-1	1
SNVT_pump_sensor.gas_brake_active	boolean_t	1	none (enumeration)
	-1	-1	1

4.153 SNVT_abs_humid

Overview: Absolute humidity in gram/kilogram.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
160	2	unsigned long	0	65 535	$1* 10^{-2} *$ (Raw+0)	0,01 g/kg

4.154 SNVT_flow_p

Overview: Flow volume in cubic meters/hour.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
161	2	unsigned long	0	65 535	$1* 10^{-2} *$ (Raw+0)	0,01 m^3/h

4.155 SNVT_dev_c_mode

Overview: Device control mode.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
162	1	device_c_mode_t	-1	32	none (enumeration)	-1

4.156 SNVT_valve_mode

Overview: Valve mode.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
163	1	valve_mode_t	-1	7	none (enumeration)	-1

4.157 SNVT_alarm_2

Overview: Alarm status 2. Used to report alarm status for a functional block or device. Replaces SNVT_alarm.

Index	Size in Bytes	CNP Data Type			
164	31	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_alarm_2.alarm_type			alarm_type_t	32	none (enumeration)
			-1	-13	1
SNVT_alarm_2.priority_level			priority_level_t	11	none (enumeration)
			-1	-1	1
SNVT_alarm_2.alarm_time			unsigned quad	4 294 967 295	$1* 10^0$ * (Raw+0)
			4 294 967 295	0	1 s
SNVT_alarm_2.milliseconds			signed long	999	$1* 10^0$ * (Raw+0)
			-1	-1	1 ms
SNVT_alarm_2.sequence_number			unsigned short	255	$1* 10^0$ * (Raw+0)
			(none)	0	1 count
SNVT_alarm_2.description			unsigned char	126	$1* 10^0$ * (Raw+0)
			(none)	32	1 array of 22 characters

4.158 SNVT_state_64

Overview: State vector. Each state is a boolean single-bit value.

Index	Size in Bytes	CNP Data Type			
165	8	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_state_64.bit0			bitfield	1	$1* 10^0$ * (Raw+0)
				0	1 (bits: 1) (offset: 0)
SNVT_state_64.bit1			bitfield	1	$1* 10^0$ * (Raw+0)

		0	1 (bits: 1) (offset: 1)
SNVT_state_64.bit2	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 2)
SNVT_state_64.bit3	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 3)
SNVT_state_64.bit4	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 4)
SNVT_state_64.bit5	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 5)
SNVT_state_64.bit6	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 6)
SNVT_state_64.bit7	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 7)
SNVT_state_64.bit8	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 0)
SNVT_state_64.bit9	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 1)
SNVT_state_64.bit10	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 2)
SNVT_state_64.bit11	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 3)
SNVT_state_64.bit12	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 4)
SNVT_state_64.bit13	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 5)
SNVT_state_64.bit14	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 6)

SNVT_state_64.bit15	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 7)
SNVT_state_64.bit16	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 0)
SNVT_state_64.bit17	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 1)
SNVT_state_64.bit18	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 2)
SNVT_state_64.bit19	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 3)
SNVT_state_64.bit20	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 4)
SNVT_state_64.bit21	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 5)
SNVT_state_64.bit22	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 6)
SNVT_state_64.bit23	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 7)
SNVT_state_64.bit24	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 0)
SNVT_state_64.bit25	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 1)
SNVT_state_64.bit26	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 2)
SNVT_state_64.bit27	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 3)
SNVT_state_64.bit28	bitfield	1	1* 10^0 * (Raw+0)

		0	1 (bits: 1) (offset: 4)
SNVT_state_64.bit29	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 5)
SNVT_state_64.bit30	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 6)
SNVT_state_64.bit31	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 7)
SNVT_state_64.bit32	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 0)
SNVT_state_64.bit33	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 1)
SNVT_state_64.bit34	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 2)
SNVT_state_64.bit35	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 3)
SNVT_state_64.bit36	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 4)
SNVT_state_64.bit37	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 5)
SNVT_state_64.bit38	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 6)
SNVT_state_64.bit39	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 7)
SNVT_state_64.bit40	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 0)
SNVT_state_64.bit41	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 1)

SNVT_state_64.bit42	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 2)
SNVT_state_64.bit43	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 3)
SNVT_state_64.bit44	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 4)
SNVT_state_64.bit45	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 5)
SNVT_state_64.bit46	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 6)
SNVT_state_64.bit47	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 7)
SNVT_state_64.bit48	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 0)
SNVT_state_64.bit49	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 1)
SNVT_state_64.bit50	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 2)
SNVT_state_64.bit51	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 3)
SNVT_state_64.bit52	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 4)
SNVT_state_64.bit53	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 5)
SNVT_state_64.bit54	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 6)

SNVT_state_64.bit55	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 7)
SNVT_state_64.bit56	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 0)
SNVT_state_64.bit57	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 1)
SNVT_state_64.bit58	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 2)
SNVT_state_64.bit59	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 3)
SNVT_state_64.bit60	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 4)
SNVT_state_64.bit61	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 5)
SNVT_state_64.bit62	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 6)
SNVT_state_64.bit63	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 7)

4.159 SNVT_nv_type

Overview: Network variable type. Network variable type description for network variables that support changeable types

Index	**Size in Bytes**	**CNP Data Type**			
166	19	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_nv_type.type_program_ID			unsigned short	255	1* 10^0 * (Raw+0)
			(none)	0	1

SNVT_nv_type.type_scope	unsigned short	6	$1* 10^{0} * (Raw+0)$
	(none)	0	1 file scope
SNVT_nv_type.type_index	unsigned long	65 535	$1* 10^{0} * (Raw+0)$
	0	1	1 type index
SNVT_nv_type.type_category	nv_type_category_t	14	none (enumeration)
	-1	-1	1
SNVT_nv_type.type_length	unsigned short	31	$1* 10^{0} * (Raw+0)$
	0	1	1 bytes
SNVT_nv_type.scaling_factor_a	signed long	32 767	$1* 10^{0} * (Raw+0)$
	32 767	-32 768	1
SNVT_nv_type.scaling_factor_b	signed long	32 767	$1* 10^{0} * (Raw+0)$
	32 767	-32 768	1
SNVT_nv_type.scaling_factor_c	signed long	32 767	$1* 10^{0} * (Raw+0)$
	32 767	-32 768	1

4.160 SNVT_ent_opmode

Overview: Entry operation mode. used to send operation-mode information to an entry object, e.g., a door, lock, sluice, or something which allows/prohibits entry to an area

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
168	1	ent_opmode_cmd_t	-1	16	none (enumeration)	-1

4.161 SNVT_ent_state

Overview: Entry state. Desired state for an entry object, e.g., a door, lock, sluice, or something that controls entry of an area

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
169	1	ent_cmd_t	-1	13	none (enumeration)	-1

4.162 SNVT_ent_status

Overview: Entry status. Status information from an entry object, e.g., a door, lock, sluice, or something that allows/prohibits entry into an area

Index	Size in Bytes	CNP Data Type			
170	5	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_ent_status.unlocked			bitfield	1	1* 10^0 * (Raw+0)
				0	1 (bits: 1) (offset: 0)
SNVT_ent_status.locked			bitfield	1	1* 10^0 * (Raw+0)
				0	1 (bits: 1) (offset: 1)
SNVT_ent_status.security_locked			bitfield	1	1* 10^0 * (Raw+0)
				0	1 (bits: 1) (offset: 2)
SNVT_ent_status.closed			bitfield	1	1* 10^0 * (Raw+0)
				0	1 (bits: 1) (offset: 3)
SNVT_ent_status.open			bitfield	1	1* 10^0 * (Raw+0)
				0	1 (bits: 1) (offset: 4)
SNVT_ent_status.in_alarm			bitfield	1	1* 10^0 * (Raw+0)
				0	1 (bits: 1) (offset: 5)
SNVT_ent_status.in_error_cond			bitfield	1	1* 10^0 * (Raw+0)
				0	1 (bits: 1) (offset: 6)
SNVT_ent_status.open_pre_alarm			bitfield	1	1* 10^0 * (Raw+0)
				0	1 (bits: 1) (offset: 7)
SNVT_ent_status.open_alarm			bitfield	1	1* 10^0 * (Raw+0)
				0	1 (bits: 1) (offset: 0)
SNVT_ent_status.service_alarm			bitfield	1	1* 10^0 * (Raw+0)
				0	1 (bits: 1) (offset: 1)

SNVT_ent_status.tamper	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 2)
SNVT_ent_status.entry_req	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 3)
SNVT_ent_status.exit_req	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 4)
SNVT_ent_status.key_req	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 5)
SNVT_ent_status.safety_ext_req	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 6)
SNVT_ent_status.emergency_req	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 7)
SNVT_ent_status.unable_lock	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 0)
SNVT_ent_status.unable_unlock	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 1)
SNVT_ent_status.stuck	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 2)
SNVT_ent_status.forced_open	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 3)
SNVT_ent_status.forced_close	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 4)
SNVT_ent_status.opening	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 5)
SNVT_ent_status.closing	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 6)

SNVT_ent_status.moving	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 7)
SNVT_ent_status.stopped	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 0)
SNVT_ent_status.safety_alarm	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 1)
SNVT_ent_status.unknown_state	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 2)
SNVT_ent_status.cmd_fb	ent_opmode_cmd_t	16	none (enumeration)
	-1	-1	1

4.163 SNVT_flow_dir

Overview: Flow direction. Direction of allowable flow, or direction of present flow.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
171	1	flow_direction_t	-1	3	none (enumeration)	-1

4.164 SNVT_hvac_satsts

Overview: HVAC saturation status. A value of 0 in a field indicates that the resource associated with that field has not saturated or reached an end stop before attaining the required setpoint. A value of 1 indicates that the resource associated with that field has saturated or reached an end stop without attaining the required setpoint.

Index	Size in Bytes	CNP Data Type			
172	2	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_hvac_satsts.pri_heat			bitfield	1	1* 10^0 * (Raw+0)
				0	1 (bits: 1) (offset: 0)
SNVT_hvac_satsts.sec_heat			bitfield	1	1* 10^0 * (Raw+0)
				0	1 (bits: 1) (offset: 1)
SNVT_hvac_satsts.pri_cool			bitfield	1	1* 10^0 * (Raw+0)

		0	1 (bits: 1) (offset: 2)
SNVT_hvac_satsts.sec_cool	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 3)
SNVT_hvac_satsts.pri_duct_starved	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 4)
SNVT_hvac_satsts.sec_duct_starved	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 5)
SNVT_hvac_satsts.reserved	bitfield	3	1* 10^0 * (Raw+0)
		0	1 (bits: 2) (offset: 6)
SNVT_hvac_satsts.reserved1	bitfield	15	1* 10^0 * (Raw+0)
		0	1 (bits: 4) (offset: 0)
SNVT_hvac_satsts.manufacturer_defined	bitfield	15	1* 10^0 * (Raw+0)
		0	1 (bits: 4) (offset: 4)

4.165 SNVT_dev_status

Overview: Device status. Status of the device

Index	**Size in Bytes**	**CNP Data Type**			
173	4	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_dev_status.device_select			device_select_t	1	none (enumeration)
			-1	-1	1
SNVT_dev_status.dev_type			union		
SNVT_dev_status.dev_type.pump_ctrl			structure		
SNVT_dev_status.dev_type.pump_ctrl.device_fault			bitfield	1	1* 10^0 * (Raw+0)
				0	1 (bits: 1) (offset: 0)

SNVT_dev_status.dev_type.pump_ctrl.supply_fault	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 1)
SNVT_dev_status.dev_type.pump_ctrl.reserved1_2	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 2)
SNVT_dev_status.dev_type.pump_ctrl.speed_low	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 3)
SNVT_dev_status.dev_type.pump_ctrl.speed_high	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 4)
SNVT_dev_status.dev_type.pump_ctrl.reserved1_5	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 5)
SNVT_dev_status.dev_type.pump_ctrl.setpt_out_of_range	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 6)
SNVT_dev_status.dev_type.pump_ctrl.reserved1_7	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 7)
SNVT_dev_status.dev_type.pump_ctrl.local_control	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 0)
SNVT_dev_status.dev_type.pump_ctrl.reserved2_1	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 1)
SNVT_dev_status.dev_type.pump_ctrl.running	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 2)
SNVT_dev_status.dev_type.pump_ctrl.reserved2_3	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 3)
SNVT_dev_status.dev_type.pump_ctrl.remote_press	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 4)
SNVT_dev_status.dev_type.pump_ctrl.remote_flow	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 5)

SNVT_dev_status.dev_type.pump_ctrl.remote_temp	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 6)
SNVT_dev_status.dev_type.pump_ctrl.reserved2_7	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 7)
SNVT_dev_status.dev_type.pump_ctrl.reserved3_0_7	bitfield	255	1* 10^0 * (Raw+0)
		0	1 (bits: 8) (offset: 0)
SNVT_dev_status.dev_type.valve_pos	structure		
SNVT_dev_status.dev_type.valve_pos.running	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 0)
SNVT_dev_status.dev_type.valve_pos.adapting	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 1)
SNVT_dev_status.dev_type.valve_pos.initializing	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 2)
SNVT_dev_status.dev_type.valve_pos.local_control	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 3)
SNVT_dev_status.dev_type.valve_pos.setpt_out_of_range	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 4)
SNVT_dev_status.dev_type.valve_pos.remote_ctrl_signal	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 5)
SNVT_dev_status.dev_type.valve_pos.reserved1_6_7	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 2) (offset: 6)
SNVT_dev_status.dev_type.valve_pos.hw_emergency	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 0)
SNVT_dev_status.dev_type.valve_pos.sw_emergency	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 1)

SNVT_dev_status.dev_type.valve_pos.reserved2_2_7	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 6) (offset: 2)
SNVT_dev_status.dev_type.valve_pos.reserved3_0_7	bitfield	255	1* 10^0 * (Raw+0)
		0	1 (bits: 8) (offset: 0)

4.166 SNVT_dev_fault

Overview: Device fault states. Fault information for the device

Index	**Size in Bytes**	**CNP Data Type**			
174	4	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_dev_fault.device_select			device_select_t	1	none (enumeration)
			-1	-1	1
SNVT_dev_fault.dev_type			union		
SNVT_dev_fault.dev_type.pump_ctrl			structure		
SNVT_dev_fault.dev_type.pump_ctrl.sf_voltage_low			bitfield	1	1* 10^0 * (Raw+0)
				0	1 (bits: 1) (offset: 0)
SNVT_dev_fault.dev_type.pump_ctrl.sf_voltage_high			bitfield	1	1* 10^0 * (Raw+0)
				0	1 (bits: 1) (offset: 1)
SNVT_dev_fault.dev_type.pump_ctrl.sf_phase			bitfield	1	1* 10^0 * (Raw+0)
				0	1 (bits: 1) (offset: 2)
SNVT_dev_fault.dev_type.pump_ctrl.sf_no_fluid			bitfield	1	1* 10^0 * (Raw+0)
				0	1 (bits: 1) (offset: 3)
SNVT_dev_fault.dev_type.pump_ctrl.sf_press_low			bitfield	1	1* 10^0 * (Raw+0)
				0	1 (bits: 1) (offset: 4)

SNVT_dev_fault.dev_type.pump_ctrl.sf_press_high	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 5)
SNVT_dev_fault.dev_type.pump_ctrl.sf_general_fault	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 6)
SNVT_dev_fault.dev_type.pump_ctrl.sf_reserved1_7	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 7)
SNVT_dev_fault.dev_type.pump_ctrl.df_motor_temp	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 0)
SNVT_dev_fault.dev_type.pump_ctrl.df_motor_failure	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 1)
SNVT_dev_fault.dev_type.pump_ctrl.df_pump_blocked	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 2)
SNVT_dev_fault.dev_type.pump_ctrl.df_elect_temp	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 3)
SNVT_dev_fault.dev_type.pump_ctrl.df_elect_failure_nf	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 4)
SNVT_dev_fault.dev_type.pump_ctrl.df_elect_failure	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 5)
SNVT_dev_fault.dev_type.pump_ctrl.df_sensor_failure	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 6)
SNVT_dev_fault.dev_type.pump_ctrl.df_general_fault	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 7)
SNVT_dev_fault.dev_type.pump_ctrl.reserved3_0_7	bitfield	255	1* 10^0 * (Raw+0)
		0	1 (bits: 8) (offset: 0)
SNVT_dev_fault.dev_type.valve_pos	structure		
SNVT_dev_fault.dev_type.valve_pos.df_valve_blocked	bitfield	1	1* 10^0 * (Raw+0)

		0	1 (bits: 1) (offset: 0)
SNVT_dev_fault.dev_type.valve_pos.df_blocked_direction_open	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 1)
SNVT_dev_fault.dev_type.valve_pos.df_blocked_direction_close	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 2)
SNVT_dev_fault.dev_type.valve_pos.df_position_error	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 3)
SNVT_dev_fault.dev_type.valve_pos.df_stroke_out_of_range	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 4)
SNVT_dev_fault.dev_type.valve_pos.df_initialization	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 5)
SNVT_dev_fault.dev_type.valve_pos.df_vibration_cavitation	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 6)
SNVT_dev_fault.dev_type.valve_pos.df_ed_too_high	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 7)
SNVT_dev_fault.dev_type.valve_pos.reserved1_0_2	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 3) (offset: 0)
SNVT_dev_fault.dev_type.valve_pos.ee_oscillating	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 3)
SNVT_dev_fault.dev_type.valve_pos.ee_valve_too_large	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 4)
SNVT_dev_fault.dev_type.valve_pos.ee_valve_too_small	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 5)
SNVT_dev_fault.dev_type.valve_pos.reserved2_6_7	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 2) (offset: 6)
SNVT_dev_fault.dev_type.valve_pos.reserved3_0_7	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 0)

SNVT_dev_fault.dev_type.valve_pos.sf_voltage_out_of_range	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 1)
SNVT_dev_fault.dev_type.valve_pos.sf_electronic_high_temp	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 2)
SNVT_dev_fault.dev_type.valve_pos.sf_frictional_resistance	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 3)
SNVT_dev_fault.dev_type.valve_pos.reserved4_4_6	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 3) (offset: 4)
SNVT_dev_fault.dev_type.valve_pos.general_fault	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 7)

4.167 SNVT_dev_maint

Overview: Device maintenance. Device-maintenance states

Index	**Size in Bytes**	**CNP Data Type**			
175	4	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_dev_maint.device_select			device_select_t	1	none (enumeration)
			-1	-1	1
SNVT_dev_maint.dev_type			union		
SNVT_dev_maint.dev_type.pump_ctrl			structure		
SNVT_dev_maint.dev_type.pump_ctrl.service_required			bitfield	1	1* 10^0 * (Raw+0)
				0	1 (bits: 1) (offset: 0)
SNVT_dev_maint.dev_type.pump_ctrl.bearings_change			bitfield	1	1* 10^0 * (Raw+0)
				0	1 (bits: 1) (offset: 1)
SNVT_dev_maint.dev_type.pump_ctrl.bearings_lubricate			bitfield	1	1* 10^0 * (Raw+0)

		0	1 (bits: 1) (offset: 2)
SNVT_dev_maint.dev_type.pump_ctrl.shaftseal_change	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 3)
SNVT_dev_maint.dev_type.pump_ctrl.reserved1_4_7	bitfield	15	1* 10^0 * (Raw+0)
		0	1 (bits: 4) (offset: 4)
SNVT_dev_maint.dev_type.pump_ctrl.reserved2_0_7	bitfield	255	1* 10^0 * (Raw+0)
		0	1 (bits: 8) (offset: 0)
SNVT_dev_maint.dev_type.pump_ctrl.reserved3_0_7	bitfield	255	1* 10^0 * (Raw+0)
		0	1 (bits: 8) (offset: 0)
SNVT_dev_maint.dev_type.valve_pos	structure		
SNVT_dev_maint.dev_type.valve_pos.motor_maint	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 0)
SNVT_dev_maint.dev_type.valve_pos.packing_change	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 1)
SNVT_dev_maint.dev_type.valve_pos.electronics_check	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 2)
SNVT_dev_maint.dev_type.valve_pos.positioning_check	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 3)
SNVT_dev_maint.dev_type.valve_pos.lubrication_check	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 4)
SNVT_dev_maint.dev_type.valve_pos.return_check	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 5)
SNVT_dev_maint.dev_type.valve_pos.battery_check	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 6)
SNVT_dev_maint.dev_type.valve_pos.reserved1_7	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 7)

SNVT_dev_maint.dev_type.valve_pos.reserved2_0_7	bitfield	255	1* 10^0 * (Raw+0)
		0	1 (bits: 8) (offset: 0)
SNVT_dev_maint.dev_type.valve_pos.reserved3_0_6	bitfield	127	1* 10^0 * (Raw+0)
		0	1 (bits: 7) (offset: 0)
SNVT_dev_maint.dev_type.valve_pos.general_maint	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 7)

4.168 SNVT_date_event

Overview: Date event. Reports the status of a schedule

Index	Size in Bytes	CNP Data Type			
176	26	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_date_event.days_to_active			signed long	32 767	1* 10^0 * (Raw+0)
			32 767	-32 768	1 day
SNVT_date_event.days_to_inactive			signed long	32 767	1* 10^0 * (Raw+0)
			-32 768	-32 768	1 day
SNVT_date_event.name			unsigned char	126	1* 10^0 * (Raw+0)
			(none)	32	1 array of 22 characters

4.169 SNVT_sched_val

Overview: Scheduler value. Index from scheduler that selects entry in SCPTvalueDefinition array, or is a direct value output

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
177	1	unsigned short	0	255	1* 10^0 * (Raw+0)	1

4.170 SNVT_sec_state

Overview: Security State. It is used to communicate with security devices and is used for control of devices

Index	Size in Bytes	CNP Data Type	
178	3	structure	

Field	CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_sec_state.state	sec_state_t	19	none (enumeration)
	-1	-1	1
SNVT_sec_state.identity	unsigned long	65 535	$1* 10^0 *$ (Raw+0)
	(none)	0	1

4.171 SNVT_sec_status

Overview: Security Status. Status information from a security device.

Index	Size in Bytes	CNP Data Type			
179	3	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_sec_status.status			sec_status_t	25	none (enumeration)
			-1	-1	1
SNVT_sec_status.identity			unsigned long	65 535	$1* 10^0 *$ (Raw+0)
			(none)	0	1

4.172 SNVT_sblnd_state

Overview: Sunblind State. Provides the present state of a sunblind

Index	Size in Bytes	CNP Data Type			
180	6	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_sblnd_state.pos			SNVT_setting		
SNVT_sblnd_state.cmd_source			sblnd_cmd_source_t	127	none (enumeration)
			-1	-1	1
SNVT_sblnd_state.error_code			sblnd_error_t	14	none (enumeration)
			-1	-1	1

4.173 SNVT_rac_ctrl

Overview: Rail-Audio Controller Control. Invokes audio control for a given source

Index	Size in Bytes	CNP Data Type			
181	18	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_rac_ctrl.audio_line			bitfield	7	1* 10^0 * (Raw+0)
				0	1 (bits: 3) (offset: 0)
SNVT_rac_ctrl.duplex_full			bitfield	1	1* 10^0 * (Raw+0)
				0	1 (bits: 1) (offset: 3)
SNVT_rac_ctrl.dest_p2p			bitfield	1	1* 10^0 * (Raw+0)
				0	1 (bits: 1) (offset: 4)
SNVT_rac_ctrl.reserved			bitfield	7	1* 10^0 * (Raw+0)
				0	1 (bits: 3) (offset: 5)

SNVT_rac_ctrl.audio_type	rail_audio_type_t	17	none (enumeration)
	-1	-1	1
SNVT_rac_ctrl.addr_init	structure		
SNVT_rac_ctrl.addr_init.unit_id	bitfield	8	1* 10^0 * (Raw+0)
		0	1 (bits: 4) (offset: 0)
SNVT_rac_ctrl.addr_init.location	bitfield	15	1* 10^0 * (Raw+0)
		0	1 (bits: 4) (offset: 4)
SNVT_rac_ctrl.addr_init.car_id	bitfield	31	1* 10^0 * (Raw+0)
		0	1 (bits: 5) (offset: 0)
SNVT_rac_ctrl.addr_init.reserved	bitfield	7	1* 10^0 * (Raw+0)
		0	1 (bits: 3) (offset: 5)
SNVT_rac_ctrl.addr_init.audio_sensor_type	rail_audio_sensor_type_t	23	none (enumeration)
	-1	-1	1
SNVT_rac_ctrl.addr_talk	structure		
SNVT_rac_ctrl.addr_talk.unit_id	bitfield	15	1* 10^0 * (Raw+0)
		0	1 (bits: 4) (offset: 0)
SNVT_rac_ctrl.addr_talk.location	bitfield	15	1* 10^0 * (Raw+0)
		0	1 (bits: 4) (offset: 4)
SNVT_rac_ctrl.addr_talk.car_id	bitfield	31	1* 10^0 * (Raw+0)
		0	1 (bits: 5) (offset: 0)
SNVT_rac_ctrl.addr_talk.reserved	bitfield	7	1* 10^0 * (Raw+0)
		0	1 (bits: 3) (offset: 5)
SNVT_rac_ctrl.addr_talk.audio_sensor_type	rail_audio_sensor_type_t	23	none (enumeration)
	-1	-1	1
SNVT_rac_ctrl.addr_dest	union		

SNVT_rac_ctrl.addr_dest.p2p	structure		
SNVT_rac_ctrl.addr_dest.p2p.unit_id	bitfield	15	1* 10^0 * (Raw+0)
		0	1 (bits: 4) (offset: 0)
SNVT_rac_ctrl.addr_dest.p2p.location	bitfield	15	1* 10^0 * (Raw+0)
		0	1 (bits: 4) (offset: 4)
SNVT_rac_ctrl.addr_dest.p2p.car_id	bitfield	31	1* 10^0 * (Raw+0)
		0	1 (bits: 5) (offset: 0)
SNVT_rac_ctrl.addr_dest.p2p.reserved	bitfield	7	1* 10^0 * (Raw+0)
		0	1 (bits: 3) (offset: 5)
SNVT_rac_ctrl.addr_dest.p2p.audio_sensor_type	rail_audio_sensor_type_t	23	none (enumeration)
	-1	-1	1
SNVT_rac_ctrl.addr_dest.p2m	structure		
SNVT_rac_ctrl.addr_dest.p2m.mask_unit	unsigned char	255	1* 10^0 * (Raw+0)
	(none)	0	1
SNVT_rac_ctrl.addr_dest.p2m.mask_car	unsigned char	255	1* 10^0 * (Raw+0)
	(none)	0	1
SNVT_rac_ctrl.addr_dest.p2m.mask_location	unsigned char	255	1* 10^0 * (Raw+0)
	(none)	0	1
SNVT_rac_ctrl.addr_dest.p2m.mask_audio	unsigned char	255	1* 10^0 * (Raw+0)
	(none)	0	1

4.174 SNVT_rac_req

Overview: Rail-Audio Controller Request. Requests audio control for a given source

Index	Size in Bytes	CNP Data Type	
182	15	structure	

Field	CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_rac_req.dest_def	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 0)
SNVT_rac_req.dest_p2p	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 1)
SNVT_rac_req.reserved	bitfield	63	1* 10^0 * (Raw+0)
		0	1 (bits: 6) (offset: 2)
SNVT_rac_req.audio_type	rail_audio_type_t	17	none (enumeration)
	-1	-1	1
SNVT_rac_req.addr_init	structure		
SNVT_rac_req.addr_init.unit_id	bitfield	15	1* 10^0 * (Raw+0)
		0	1 (bits: 4) (offset: 0)
SNVT_rac_req.addr_init.location	bitfield	15	1* 10^0 * (Raw+0)
		0	1 (bits: 4) (offset: 4)
SNVT_rac_req.addr_init.car_id	bitfield	31	1* 10^0 * (Raw+0)
		0	1 (bits: 5) (offset: 0)
SNVT_rac_req.addr_init.reserved	bitfield	7	1* 10^0 * (Raw+0)
		0	1 (bits: 3) (offset: 5)
SNVT_rac_req.addr_init.audio_sensor_type	rail_audio_sensor_type_t	23	none (enumeration)
	-1	-1	1
SNVT_rac_req.addr_dest	union		
SNVT_rac_req.addr_dest.p2p	structure		
SNVT_rac_req.addr_dest.p2p.unit_id	bitfield	15	1* 10^0 * (Raw+0)

		0	1 (bits: 4) (offset: 0)
SNVT_rac_req.addr_dest.p2p.location	bitfield	15	1* 10^0 * (Raw+0)
		0	1 (bits: 4) (offset: 4)
SNVT_rac_req.addr_dest.p2p.car_id	bitfield	31	1* 10^0 * (Raw+0)
		0	1 (bits: 5) (offset: 0)
SNVT_rac_req.addr_dest.p2p.reserved	bitfield	7	1* 10^0 * (Raw+0)
		0	1 (bits: 3) (offset: 5)
SNVT_rac_req.addr_dest.p2p.audio_sensor_type	rail_audio_sensor_type_t	23	none (enumeration)
	-1	-1	1
SNVT_rac_req.addr_dest.p2m	structure		
SNVT_rac_req.addr_dest.p2m.mask_unit	unsigned char	255	1* 10^0 * (Raw+0)
	(none)	0	1
SNVT_rac_req.addr_dest.p2m.mask_car	unsigned char	255	1* 10^0 * (Raw+0)
	(none)	0	1
SNVT_rac_req.addr_dest.p2m.mask_location	unsigned char	255	1* 10^0 * (Raw+0)
	(none)	0	1
SNVT_rac_req.addr_dest.p2m.mask_audio	unsigned char	255	1* 10^0 * (Raw+0)
	(none)	0	1

4.175 SNVT_count_32

Overview: Absolute count. A 32-bit counter

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
183	4	unsigned quad	0	4 294 967 295	1* 10^0 * (Raw+0)	1

4.176 SNVT_clothes_w_c

Overview: Clothes Washer Command. Used to program and start a clothes washer

Index	Size in Bytes	CNP Data Type	
184	21	structure	

Field	CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_clothes_w_c.cycle	appl_cwc_t	3	none (enumeration)
	-1	-1	1
SNVT_clothes_w_c.subcycle	appl_cws_t	12	none (enumeration)
	-1	-1	1
SNVT_clothes_w_c.rervd	unsigned short	255	$1* 10^0 * (Raw+0)$
	(none)	0	1
SNVT_clothes_w_c.action	structure		
SNVT_clothes_w_c.action.power_on	bitfield	1	$1* 10^0 * (Raw+0)$
		0	1 (bits: 1) (offset: 0)
SNVT_clothes_w_c.action.run_mode	bitfield	1	$1* 10^0 * (Raw+0)$
		0	1 (bits: 1) (offset: 1)
SNVT_clothes_w_c.action.rsrvd2_7	bitfield	63	$1* 10^0 * (Raw+0)$
		0	1 (bits: 6) (offset: 2)
SNVT_clothes_w_c.function	structure		
SNVT_clothes_w_c.function.program	appl_cwp_t	11	none (enumeration)
	-1	-1	1
SNVT_clothes_w_c.function.wash	structure		
SNVT_clothes_w_c.function.wash.load_level	discrete_levels_t	4	none (enumeration)
	-1	-1	1
SNVT_clothes_w_c.function.wash.temp	unsigned short	255	$1* 10^0 * (Raw+0)$
	(none)	0	1
SNVT_clothes_w_c.function.wash.time	unsigned short	255	$1* 10^0 * (Raw+0)$

	(none)	0	1
SNVT_clothes_w_c.function.wash.prewash	boolean_t	1	none (enumeration)
	-1	-1	1
SNVT_clothes_w_c.function.rinse	structure		
SNVT_clothes_w_c.function.rinse.temp	unsigned short	255	$1* 10^0 *$ (Raw+0)
	(none)	0	1
SNVT_clothes_w_c.function.rinse.repeat	unsigned short	9	$1* 10^0 *$ (Raw+0)
	(none)	0	1
SNVT_clothes_w_c.function.rinse.option	appl_rin_t	3	none (enumeration)
	-1	-1	1
SNVT_clothes_w_c.function.spin	structure		
SNVT_clothes_w_c.function.spin.speed	SNVT_rpm		
SNVT_clothes_w_c.function.spin.time	unsigned short	255	$1* 10^0 *$ (Raw+0)
	(none)	0	1
SNVT_clothes_w_c.function.spin.hold	boolean_t	1	none (enumeration)
	-1	-1	1
SNVT_clothes_w_c.function.dry	structure		
SNVT_clothes_w_c.function.dry.temp	unsigned short	1	$1* 10^0 *$ (Raw+0)
	(none)	0	1
SNVT_clothes_w_c.function.dry.duration	structure		
SNVT_clothes_w_c.function.dry.duration.time	unsigned short	255	$1* 10^0 *$ (Raw+0)
	(none)	0	1

SNVT_clothes_w_c.function.dry.duration.dryness	discrete_levels_t	4	none (enumeration)
	-1	-1	1
SNVT_clothes_w_c.time_remaining	SNVT_time_min		

4.177 SNVT_clothes_w_m

Overview: Clothes Washer-Management Status. Provides status of door/lid and drain

Index	Size in Bytes	CNP Data Type			
185	1	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_clothes_w_m.door_ajar			bitfield	1	1* 10^0 * (Raw+0)
				0	1 (bits: 1) (offset: 0)
SNVT_clothes_w_m.drain_on			bitfield	1	1* 10^0 * (Raw+0)
				0	1 (bits: 1) (offset: 1)
SNVT_clothes_w_m.reserved			bitfield	63	1* 10^0 * (Raw+0)
				0	1 (bits: 6) (offset: 2)

4.178 SNVT_clothes_w_s

Overview: Clothes Washer Status. Used to provide present status from a clothes washer, including command and alarm information

Index	Size in Bytes	CNP Data Type			
186	31	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_clothes_w_s.cycle			appl_cwc_t	3	none (enumeration)
			-1	-1	1
SNVT_clothes_w_s.subcycle			appl_cws_t	12	none (enumeration)
			-1	-1	1

SNVT_clothes_w_s.washer_command_data	SNVT_clothes_w_c		
SNVT_clothes_w_s.time_remaining	SNVT_time_min		
SNVT_clothes_w_s.alarm	structure		
SNVT_clothes_w_s.alarm.alarm_reset	bitfield	1	$1* 10^0 * (Raw+0)$
		0	1 (bits: 1) (offset: 0)
SNVT_clothes_w_s.alarm.war_water_supply	bitfield	1	$1* 10^0 * (Raw+0)$
		0	1 (bits: 1) (offset: 1)
SNVT_clothes_w_s.alarm.war_drain_slow	bitfield	1	$1* 10^0 * (Raw+0)$
		0	1 (bits: 1) (offset: 2)
SNVT_clothes_w_s.alarm.war_door_open	bitfield	1	$1* 10^0 * (Raw+0)$
		0	1 (bits: 1) (offset: 3)
SNVT_clothes_w_s.alarm.war_load_unbalanced	bitfield	1	$1* 10^0 * (Raw+0)$
		0	1 (bits: 1) (offset: 4)
SNVT_clothes_w_s.alarm.war_filter_cleaning	bitfield	1	$1* 10^0 * (Raw+0)$
		0	1 (bits: 1) (offset: 5)
SNVT_clothes_w_s.alarm.war_hoses_reversed	bitfield	1	$1* 10^0 * (Raw+0)$
		0	1 (bits: 1) (offset: 6)
SNVT_clothes_w_s.alarm.war_voltage_low	bitfield	1	$1* 10^0 * (Raw+0)$
		0	1 (bits: 1) (offset: 7)
SNVT_clothes_w_s.alarm.war_power_failure	bitfield	1	$1* 10^0 * (Raw+0)$
		0	1 (bits: 1) (offset: 0)
SNVT_clothes_w_s.alarm.war_drain_open	bitfield	1	$1* 10^0 * (Raw+0)$
		0	1 (bits: 1) (offset: 1)
SNVT_clothes_w_s.alarm.war_execute_fail	bitfield	1	$1* 10^0 * (Raw+0)$

		0	1 (bits: 1) (offset: 2)
SNVT_clothes_w_s.alarm.war_door_locked	bitfield	1	$1* 10^{0} * (Raw+0)$
		0	1 (bits: 1) (offset: 3)
SNVT_clothes_w_s.alarm.war_service	bitfield	1	$1* 10^{0} * (Raw+0)$
		0	1 (bits: 1) (offset: 4)
SNVT_clothes_w_s.alarm.war_rsrvd5	bitfield	1	$1* 10^{0} * (Raw+0)$
		0	1 (bits: 1) (offset: 5)
SNVT_clothes_w_s.alarm.war_rsrvd6	bitfield	1	$1* 10^{0} * (Raw+0)$
		0	1 (bits: 1) (offset: 6)
SNVT_clothes_w_s.alarm.war_rsrvd7	bitfield	1	$1* 10^{0} * (Raw+0)$
		0	1 (bits: 1) (offset: 7)
SNVT_clothes_w_s.alarm.err_motor_stall	bitfield	1	$1* 10^{0} * (Raw+0)$
		0	1 (bits: 1) (offset: 0)
SNVT_clothes_w_s.alarm.err_water_temp	bitfield	1	$1* 10^{0} * (Raw+0)$
		0	1 (bits: 1) (offset: 1)
SNVT_clothes_w_s.alarm.err_pressure	bitfield	1	$1* 10^{0} * (Raw+0)$
		0	1 (bits: 1) (offset: 2)
SNVT_clothes_w_s.alarm.err_overflow	bitfield	1	$1* 10^{0} * (Raw+0)$
		0	1 (bits: 1) (offset: 3)
SNVT_clothes_w_s.alarm.err_water_heat	bitfield	1	$1* 10^{0} * (Raw+0)$
		0	1 (bits: 1) (offset: 4)
SNVT_clothes_w_s.alarm.err_water_leak	bitfield	1	$1* 10^{0} * (Raw+0)$
		0	1 (bits: 1) (offset: 5)
SNVT_clothes_w_s.alarm.err_motor_speed	bitfield	1	$1* 10^{0} * (Raw+0)$
		0	1 (bits: 1) (offset: 6)
SNVT_clothes_w_s.alarm.err_wash_thermistor	bitfield	1	$1* 10^{0} * (Raw+0)$
		0	1 (bits: 1) (offset: 7)

SNVT_clothes_w_s.alarm.err_dry_thermistor	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 0)
SNVT_clothes_w_s.alarm.err_dry_overheat	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 1)
SNVT_clothes_w_s.alarm.err_dry_heating	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 2)
SNVT_clothes_w_s.alarm.err_dry_fan	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 3)
SNVT_clothes_w_s.alarm.err_rsrvd4	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 4)
SNVT_clothes_w_s.alarm.err_rsrvd5	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 5)
SNVT_clothes_w_s.alarm.err_rsrvd6	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 6)
SNVT_clothes_w_s.alarm.err_rsrvd7	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 7)
SNVT_clothes_w_s.alarm.err_rsrvd0_7	bitfield	255	1* 10^0 * (Raw+0)
		0	1 (bits: 8) (offset: 0)
SNVT_clothes_w_s.alarm.manuf_code	unsigned short	255	1* 10^0 * (Raw+0)
	(none)	0	1

4.179 SNVT_clothes_w_a

Overview: Clothes Washer Alarm. Used to provide alarm status for a clothes washer

Index	Size in Bytes	CNP Data Type			
187	6	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_clothes_w_a.alarm			structure		

SNVT_clothes_w_a.alarm.alarm_reset	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 0)
SNVT_clothes_w_a.alarm.war_water_supply	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 1)
SNVT_clothes_w_a.alarm.war_drain_slow	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 2)
SNVT_clothes_w_a.alarm.war_door_open	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 3)
SNVT_clothes_w_a.alarm.war_load_unbalanced	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 4)
SNVT_clothes_w_a.alarm.war_filter_cleaning	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 5)
SNVT_clothes_w_a.alarm.war_hoses_reversed	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 6)
SNVT_clothes_w_a.alarm.war_voltage_low	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 7)
SNVT_clothes_w_a.alarm.war_power_failure	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 0)
SNVT_clothes_w_a.alarm.war_drain_open	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 1)
SNVT_clothes_w_a.alarm.war_execute_fail	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 2)
SNVT_clothes_w_a.alarm.war_door_locked	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 3)
SNVT_clothes_w_a.alarm.war_service	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 4)

SNVT_clothes_w_a.alarm.rsrvd5	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 5)
SNVT_clothes_w_a.alarm.rsrvd6	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 6)
SNVT_clothes_w_a.alarm.rsrvd7	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 7)
SNVT_clothes_w_a.alarm.err_motor_stall	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 0)
SNVT_clothes_w_a.alarm.err_water_temp	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 1)
SNVT_clothes_w_a.alarm.err_pressure	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 2)
SNVT_clothes_w_a.alarm.err_overflow	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 3)
SNVT_clothes_w_a.alarm.err_water_heat	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 4)
SNVT_clothes_w_a.alarm.err_water_leak	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 5)
SNVT_clothes_w_a.alarm.err_motor_speed	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 6)
SNVT_clothes_w_a.alarm.err_wash_thermistor	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 7)
SNVT_clothes_w_a.alarm.err_dry_thermistor	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 0)
SNVT_clothes_w_a.alarm.err_dry_overheat	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 1)
SNVT_clothes_w_a.alarm.err_dry_heating	bitfield	1	1* 10^0 * (Raw+0)

		0	1 (bits: 1) (offset: 2)
SNVT_clothes_w_a.alarm.err_dry_fan	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 3)
SNVT_clothes_w_a.alarm.err_rsrvd4	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 4)
SNVT_clothes_w_a.alarm.err_rsrvd5	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 5)
SNVT_clothes_w_a.alarm.err_rsrvd6	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 6)
SNVT_clothes_w_a.alarm.err_rsrvd7	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 7)
SNVT_clothes_w_a.alarm.err_rsrvd0_7	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 8) (offset: 0)
SNVT_clothes_w_a.alarm.manuf_code	unsigned short	255	1* 10^0 * (Raw+0)
	(none)	0	1

4.180 SNVT_multiplier_s

Overview: Value multiplier

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
188	1	unsigned short	0	254	1* 10^{-2} * (Raw+0)	0,01

4.181 SNVT_switch_2

Overview: Switch with scene and setting control. An enhanced version of SNVT_switch with scene and setting controls similar to SNVT_scene and SNVT_setting.

Index	Size in Bytes	CNP Data Type		
189	3	structure		
Field		CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_switch_2.state		switch_state_t	58	none (enumeration)

	-1	-1	1
SNVT_switch_2.setting	union		
	Sets or reports the level, change, or angle for a switch		
SNVT_switch_2.setting.value	unsigned short	200	$5^* 10^{-1}$ * (Raw+0)
	255	0	0,5
SNVT_switch_2.setting.change	unsigned short	200	$5^* 10^{-1}$ * (Raw+0)
	255	0	0,5
SNVT_switch_2.setting.delay	unsigned short	254	$1^* 10^{0}$ * (Raw+0)
	255	0	1 s
SNVT_switch_2.setting.group_number	unsigned short	63	$1^* 10^{0}$ * (Raw+0)
	255	0	1
SNVT_switch_2.setting.multiplier	SNVT_multiplier_s		
SNVT_switch_2.setting.angle	signed short	90	$2^* 10^{0}$ * (Raw+0)
	-128	-90	2°
SNVT_switch_2.setting.fan_level	signed short	100	$1^* 10^{0}$ * (Raw+0)
	-128	-100	1
SNVT_switch_2. setting.button_number	unsigned short	255	$1^* 10^{0}$ * (Raw+0)
	0	0	1
SNVT_switch_2.scene_number	unsigned short	255	$1^* 10^{0}$ * (Raw+0)
	0	1	1

4.182 SNVT_color_2

Overview: Color.

Index	Size in Bytes	CNP Data Type			
190	5	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_color_2.encoding			color_encoding_t	3	none (enumeration)
			-1	-1	1
SNVT_color_2.color_value			union		
			Color value encoded as specified by the encoding field		
SNVT_color_2.color_value.CIE1931_lumen			structure		
			CIE1931 color space with Y output in lumen		
SNVT_color_2.color_value.CIE1931_lumen.x			unsigned short	148	$5^* 10^{-3}$ * (Raw+0)
			255	0	0,005 % of full level
SNVT_color_2.color_value.CIE1931_lumen.y			unsigned short	168	$5^* 10^{-3}$ * (Raw+0)
			255	0	0,005 % of full level
SNVT_color_2.color_value.CIE1931_lumen.absolute_Y			unsigned long	65 534	$1^* 10^{1}$ * (Raw+0)
			65 535	0	10 lumen

SNVT_color_2.color_value.CIE1931_percent	structure		
	CIE31 color space with Y output in percent of maximum lumen output of the lamp		
SNVT_color_2.color_value.CIE1931_percent.x	unsigned short	148	$5^* 10^{-3}$ * (Raw+0)
	255	0	0,005 % of full level
SNVT_color_2.color_value.CIE1931_percent.y	unsigned short	168	$5^* 10^{-3}$ * (Raw+0)
	255	0	0,005 % of full level
SNVT_color_2.color_value.CIE1931_percent.percent_Y	unsigned long	10 000	$1^* 10^{-2}$ * (Raw+0)
	65 535	0	0,01 % of full level
SNVT_color_2.color_value.RGB	structure		
SNVT_color_2.color_value.RGB.red	unsigned short	255	$1^* 10^{0}$ * (Raw+0)
	(none)	0	1
SNVT_color_2.color_value.RGB.green	unsigned short	255	$1^* 10^{0}$ * (Raw+0)
	(none)	0	1
SNVT_color_2.color_value.RGB.blue	unsigned short	255	$1^* 10^{0}$ * (Raw+0)
	(none)	0	1
SNVT_color_2.color_value.color_temperature	unsigned short	150	$50^* 10^{0}$ * (Raw+0)
	255	56	50 degrees Kelvin

4.183 SNVT_log_status

Overview: Log status. Reports the current status of a data log. Updated based on the cpLogNotificationThreshold value. Reports status only; alarms reported via Node Object nvoAlarm2 output. Required if the Node Object does not include an nvoLogStat output.

Index	**Size in Bytes**	**CNP Data Type**			
191	30	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_log_status.status			log_status_t	5	none (enumeration)
			-1	-1	1

SNVT_log_status.log_number	unsigned long	65 535	1* 10^0 * (Raw+0)
	0	1	1
SNVT_log_status.level	unsigned short	200	5* 10^{-1} * (Raw+0)
	255	0	0,5 %
SNVT_log_status.record_count	s32_type (signed 32-bit type)	2 147 483 647	1* 10^0 * (Raw+0)
	(none)	-2 147 483 648	1 records
SNVT_log_status.byte_count	s32_type (signed 32-bit type)	2 147 483 647	1* 10^0 * (Raw+0)
	(none)	-2 147 483 648	1 bytes
SNVT_log_status.total_record_count	s32_type (signed 32-bit type)	2 147 483 647	1* 10^0 * (Raw+0)
	(none)	-2 147 483 648	1 records
SNVT_log_status.records_since_notification	s32_type (signed 32-bit type)	2 147 483 647	1* 10^0 * (Raw+0)
	(none)	-2 147 483 648	1 records
SNVT_log_status.current_notify_time	SNVT_time_stamp_p		
SNVT_log_status.previous_notify_time	SNVT_time_stamp_p		

4.184 SNVT_time_stamp_p

Overview: Precision timestamp. Timestamp with hundredths of a second resolution.

Index	**Size in Bytes**	**CNP Data Type**			
192	5	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_time_stamp_p.second			unsigned quad	429 496 729 5	1* 10^0 * (Raw+0)
			(none)	0	1 s
SNVT_time_stamp_p.hundredths			unsigned short	99	1* 10^{-2} * (Raw+0)
			255	0	0,01 s

4.185 SNVT_log_fx_request

Overview: Log file transfer request. Requests a data log to be transferred via FTP. Shall be followed by a standard FTP request to get the data log file. Required on devices implementing the Data Logger functional profile that support data log transfer via FTP.

Index	Size in Bytes	CNP Data Type			
193	16	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_log_fx_request.requested_log			unsigned long	65 535	1* 10^0 * (Raw+0)
			0	1	1
SNVT_log_fx_request.record_count			unsigned quad	429 496 729 5	1* 10^0 * (Raw+0)
			(none)	0	1
SNVT_log_fx_request.start_time			SNVT_time_stamp_p		
SNVT_log_fx_request.end_time			SNVT_time_stamp_p		

4.186 SNVT_log_fx_status

Overview: Log file transfer status. Reports the status of a data log file transfer using FTP. Required on devices implementing the Data Logger functional profile that support data log transfer via FTP.

Index	Size in Bytes	CNP Data Type			
194	5	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_log_fx_status.requestor_subnet			unsigned short	255	1* 10^0 * (Raw+0)
			0	1	1
SNVT_log_fx_status.requestor_node			unsigned short	255	1* 10^0 * (Raw+0)
			0	1	1
SNVT_log_fx_status.log_number			unsigned long	65 535	1* 10^0 * (Raw+0)
			0	1	1

SNVT_log_fx_status.complete	unsigned short	200	5* 10^{-1} * (Raw+0)
	255	0	0,5

4.187 SNVT_log_request

Overview: Log status request. Requests the current status of a data log. Status is reported by a SNVT_log_status output.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
195	2	unsigned long	1	65 535	1* 10^{0} * (Raw+0)	1

4.188 SNVT_enthalpy_d

Overview: Enthalpy difference in kJ/kg.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
196	2	signed long	-32 768	32 766	1* 10^{-2} * (Raw+0)	0.01 kJ/kg

4.189 SNVT_amp_ac_mil

Overview: Electrical current in milliAmperes.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
197	2	unsigned long	0	65 534	1* 10^{0} * (Raw+0)	1 mA

4.190 SNVT_time_hour_p

Overview: Time hour in hours.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
198	4	unsigned quad	0	429 496 729 4	1* 10^{0} * (Raw+0)	1 h

4.191 SNVT_lamp_status

Overview: Lamp Status.

Index	Size in Bytes	CNP Data Type			
199	20	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_lamp_status.time_actual			SNVT_time_stamp_p		
SNVT_lamp_status.alarm_actual			structure		
SNVT_lamp_status.alarm_actual.lamp_current_high			bitfield	1	1* 10^0 * (Raw+0)
				0	1 (bits: 1) (offset: 0)
SNVT_lamp_status.alarm_actual.lamp_current_low			bitfield	1	1* 10^0 * (Raw+0)
				0	1 (bits: 1) (offset: 1)
SNVT_lamp_status.alarm_actual.main_current_high			bitfield	1	1* 10^0 * (Raw+0)
				0	1 (bits: 1) (offset: 2)
SNVT_lamp_status.alarm_actual.main_current_low			bitfield	1	1* 10^0 * (Raw+0)
				0	1 (bits: 1) (offset: 3)
SNVT_lamp_status.alarm_actual.lamp_voltage_high			bitfield	1	1* 10^0 * (Raw+0)
				0	1 (bits: 1) (offset: 4)
SNVT_lamp_status.alarm_actual.lamp_voltage_low			bitfield	1	1* 10^0 * (Raw+0)
				0	1 (bits: 1) (offset: 5)
SNVT_lamp_status.alarm_actual.main_voltage_high			bitfield	1	1* 10^0 * (Raw+0)
				0	1 (bits: 1) (offset:

			6)
SNVT_lamp_status.alarm_actual.main_voltage_low	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 7)
SNVT_lamp_status.alarm_actual.powerfactor_low	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 0)
SNVT_lamp_status.alarm_actual.OLC_temp_high	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 1)
SNVT_lamp_status.alarm_actual.power_high	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 2)
SNVT_lamp_status.alarm_actual.power_low	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 3)
SNVT_lamp_status.alarm_actual.relay_failure	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 4)
SNVT_lamp_status.alarm_actual.cap_failure	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 5)
SNVT_lamp_status.alarm_actual.lamp_failure	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 6)
SNVT_lamp_status.alarm_actual.ballast_failure	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 7)
SNVT_lamp_status.alarm_actual.inter_com_failure	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 0)
SNVT_lamp_status.alarm_actual.exter_com_failure	bitfield	1	1* 10^0 * (Raw+0)

		0	1 (bits: 1) (offset: 1)
SNVT_lamp_status.alarm_actual.main_volt_below_spec	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 2)
SNVT_lamp_status.alarm_actual.lamp_restart_count	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 3)
SNVT_lamp_status.alarm_actual.fading_ready	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 4)
SNVT_lamp_status.alarm_actual.ballast_temp_high	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 5)
SNVT_lamp_status.alarm_actual.digi_in_A	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 6)
SNVT_lamp_status.alarm_actual.digi_in_B	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 7)
SNVT_lamp_status.alarm_actual.bit_25_res	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 0)
SNVT_lamp_status.alarm_actual.bit_26_res	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 1)
SNVT_lamp_status.alarm_actual.bit_27_res	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 2)
SNVT_lamp_status.alarm_actual.bit_28_res	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 3)
SNVT_lamp_status.alarm_actual.bit_29_res	bitfield	1	1* 10^0 * (Raw+0)

		0	1 (bits: 1) (offset: 4)
SNVT_lamp_status.alarm_actual.bit_30_res	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 5)
SNVT_lamp_status.alarm_actual.bit_31_res	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 6)
SNVT_lamp_status.alarm_actual.bit_32_res	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 7)
SNVT_lamp_status.alarm_actual.bit_33_res	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 0)
SNVT_lamp_status.alarm_actual.bit_34_res	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 1)
SNVT_lamp_status.alarm_actual.bit_35_res	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 2)
SNVT_lamp_status.alarm_actual.bit_36_res	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 3)
SNVT_lamp_status.alarm_actual.bit_37_res	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 4)
SNVT_lamp_status.alarm_actual.bit_38_res	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 5)
SNVT_lamp_status.alarm_actual.bit_39_res	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 6)
SNVT_lamp_status.alarm_actual.bit_40_res	bitfield	1	1* 10^0 * (Raw+0)

		0	1 (bits: 1) (offset: 7)
SNVT_lamp_status.time_previous	SNVT_time_stamp_p		
SNVT_lamp_status.alarm_previous	structure		
SNVT_lamp_status.alarm_previous.lamp_current_high	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 0)
SNVT_lamp_status.alarm_previous.lamp_current_low	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 1)
SNVT_lamp_status.alarm_previous.main_current_high	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 2)
SNVT_lamp_status.alarm_previous.main_current_low	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 3)
SNVT_lamp_status.alarm_previous.lamp_voltage_high	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 4)
SNVT_lamp_status.alarm_previous.lamp_voltage_low	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 5)
SNVT_lamp_status.alarm_previous.main_voltage_high	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 6)
SNVT_lamp_status.alarm_previous.main_voltage_low	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 7)
SNVT_lamp_status.alarm_previous.powerfactor_low	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset:

			0)
SNVT_lamp_status.alarm_previous.OLC_temp_high	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 1)
SNVT_lamp_status.alarm_previous.power_high	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 2)
SNVT_lamp_status.alarm_previous.power_low	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 3)
SNVT_lamp_status.alarm_previous.relay_failure	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 4)
SNVT_lamp_status.alarm_previous.cap_failure	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 5)
SNVT_lamp_status.alarm_previous.lamp_failure	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 6)
SNVT_lamp_status.alarm_previous.ballast_failure	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 7)
SNVT_lamp_status.alarm_previous.inter_com_failure	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 0)
SNVT_lamp_status.alarm_previous.exter_com_failure	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 1)
SNVT_lamp_status.alarm_previous.main_volt_below_spec	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 2)
SNVT_lamp_status.alarm_previous.lamp_restart_count	bitfield	1	1* 10^0 * (Raw+0)

		0	1 (bits: 1) (offset: 3)
SNVT_lamp_status.alarm_previous.fading_ready	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 4)
SNVT_lamp_status.alarm_previous.ballast_temp_high	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 5)
SNVT_lamp_status.alarm_previous.digi_in_A	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 6)
SNVT_lamp_status.alarm_previous.digi_in_B	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 7)
SNVT_lamp_status.alarm_previous.bit_25_res	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 0)
SNVT_lamp_status.alarm_previous.bit_26_res	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 1)
SNVT_lamp_status.alarm_previous.bit_27_res	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 2)
SNVT_lamp_status.alarm_previous.bit_28_res	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 3)
SNVT_lamp_status.alarm_previous.bit_29_res	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 4)
SNVT_lamp_status.alarm_previous.bit_30_res	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 5)
SNVT_lamp_status.alarm_previous.bit_31_res	bitfield	1	1* 10^0 * (Raw+0)

		0	1 (bits: 1) (offset: 6)
SNVT_lamp_status.alarm_previous.bit_32_res	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 7)
SNVT_lamp_status.alarm_previous.bit_33_res	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 0)
SNVT_lamp_status.alarm_previous.bit_34_res	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 1)
SNVT_lamp_status.alarm_previous.bit_35_res	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 2)
SNVT_lamp_status.alarm_previous.bit_36_res	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 3)
SNVT_lamp_status.alarm_previous.bit_37_res	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 4)
SNVT_lamp_status.alarm_previous.bit_38_res	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 5)
SNVT_lamp_status.alarm_previous.bit_39_res	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 6)
SNVT_lamp_status.alarm_previous.bit_40_res	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 7)

4.192 SNVT_environment

Overview: Environment.

Index	Size in Bytes	CNP Data Type	

200	22	structure	

Field	CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_environment.lampCurrent	SNVT_amp_ac_mil		
SNVT_environment.lampVoltage	SNVT_volt_ac		
SNVT_environment.supplyVoltage	SNVT_volt_ac		
SNVT_environment.supplyCurrent	SNVT_amp_ac_mil		
SNVT_environment.ballastTemp	SNVT_temp_p		
SNVT_environment.power	SNVT_power		
SNVT_environment.powerFactor	SNVT_pwr_fact		
SNVT_environment.runHours	SNVT_time_hour_p		
SNVT_environment.energy	SNVT_elec_kwh_l		

4.193 SNVT_geo_loc

Overview: Geographic Location.

Index	Size in Bytes	CNP Data Type	
201	31	structure	

Field	CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution

SNVT_geo_loc.longitude	s32_type (signed 32-bit type)	1 800 000 000	$1* 10^{-7} *$ (Raw+0)
	2147483647	-1 800 000 000	0,000 000 1
SNVT_geo_loc.latitude	s32_type (signed 32-bit type)	900 000 000	$1* 10^{-7} *$ (Raw+0)
	2147483647	-900 000 000	0,000 000 1
SNVT_geo_loc.elevation	floating-point	3,402 823 466 385 3E + 038	floating
	NAN and Out of Range	- 3,402 823 466 385 3E + 038	Elevation
SNVT_geo_loc.name	unsigned char	255	$1* 10^{0} *$ (Raw+0)
	(none)	0	1

4.194 SNVT_program_status

Overview: Program status. Indicates the current status info for the Program object, including state, runtime total, and error code with time stamp.

Index	**Size in Bytes**	**CNP Data Type**			
202	16	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_program_status.state			program_state_t	5	none (enumeration)
			-1	-1	1
SNVT_program_status.runtime			SNVT_elapsed_tm		
SNVT_program_status.last_error			program_status_error_t	63	none (enumeration)
			-1	-1	1
SNVT_program_status.time_of_last_error			SNVT_time_stamp		

4.195 SNVT_load_offsets

Overview: Load control offsets.

Index	**Size in Bytes**	**CNP Data Type**	

203	6	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_load_offsets.standby_offset			SNVT_lev_percent		
SNVT_load_offsets.standby_rotation			signed short	90	2* 10^0 * (Raw+0)
			-128	-90	2
SNVT_load_offsets.demand_response_offset			SNVT_lev_percent		
SNVT_load_offsets.demand_response_rotation			signed short	90	2* 10^0 * (Raw+0)
			-128	-90	2

4.196 SNVT_Wm2_p

Overview: Watts per square meter in W/m2.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
204	2	unsigned long	0	65 534	5* 10^{-2} * (Raw+0)	0,05 W/m^2

4.197 SNVT_safe_1

Overview: Safe protocol for 1 byte data length.

Index	Size in Bytes	CNP Data Type			
205	16	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_safe_1.id_a_header			bitfield	1	1* 10^0 * (Raw+0)
				0	1 (bits: 1) (offset: 0)
SNVT_safe_1.id_a_format			bitfield	3	1* 10^0 * (Raw+0)
				0	1 (bits: 2) (offset: 1)
SNVT_safe_1.id_a_version			bitfield	3	1* 10^0 * (Raw+0)
				0	1 (bits: 2) (offset: 3)

SNVT_safe_1.id_a_length	bitfield	7	1* 10^0 * (Raw+0)
		0	1 (bits: 3) (offset: 5)
SNVT_safe_1.address_a_1	unsigned short	255	1* 10^0 * (Raw+0)
	(none)	0	1
SNVT_safe_1.address_a_timedate	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 0)
SNVT_safe_1.address_a_2	bitfield	127	1* 10^0 * (Raw+0)
		0	1 (bits: 7) (offset: 1)
SNVT_safe_1.address_a_3	unsigned short	255	1* 10^0 * (Raw+0)
	(none)	0	1
SNVT_safe_1.time_stamp_msword	unsigned long	65 535	1* 10^0 * (Raw+0)
	(none)	0	1
SNVT_safe_1.data_a	unsigned short	255	1* 10^0 * (Raw+0)
	(none)	0	1
SNVT_safe_1.crc_a	unsigned short	255	1* 10^0 * (Raw+0)
	(none)	0	1
SNVT_safe_1.id_b_header	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 0)
SNVT_safe_1.id_b_format	bitfield	3	1* 10^0 * (Raw+0)
		0	1 (bits: 2) (offset: 1)
SNVT_safe_1.id_b_version	bitfield	3	1* 10^0 * (Raw+0)
		0	1 (bits: 2) (offset: 3)
SNVT_safe_1.id_b_length	bitfield	7	1* 10^0 * (Raw+0)
		0	1 (bits: 3) (offset: 5)
SNVT_safe_1.address_b_1	unsigned short	255	1* 10^0 * (Raw+0)
	(none)	0	1
SNVT_safe_1.address_b_timedate	bitfield	1	1* 10^0 * (Raw+0)

		0	1 (bits: 1) (offset: 0)
SNVT_safe_1.address_b_2	bitfield	127	$1* 10^0$ * (Raw+0)
		0	1 (bits: 7) (offset: 1)
SNVT_safe_1.address_b_3	unsigned short	255	$1* 10^0$ * (Raw+0)
	(none)	0	1
SNVT_safe_1.time_stamp_lsword	unsigned long	65535	$1* 10^0$ * (Raw+0)
	(none)	0	1
SNVT_safe_1.data_b	unsigned short	255	$1* 10^0$ * (Raw+0)
	(none)	0	1
SNVT_safe_1.crc_b	unsigned short	255	$1* 10^0$ * (Raw+0)
	(none)	0	1

4.198 SNVT_safe_2

Overview: Safe protocol for 2 bytes data length.

Index	Size in Bytes	CNP Data Type			
206	18	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_safe_2.id_a_header			bitfield	1	$1* 10^0$ * (Raw+0)
				0	1 (bits: 1) (offset: 0)
SNVT_safe_2.id_a_format			bitfield	3	$1* 10^0$ * (Raw+0)
				0	1 (bits: 2) (offset: 1)
SNVT_safe_2.id_a_version			bitfield	3	$1* 10^0$ * (Raw+0)
				0	1 (bits: 2) (offset: 3)
SNVT_safe_2.id_a_length			bitfield	7	$1* 10^0$ * (Raw+0)
				0	1 (bits: 3) (offset: 5)
SNVT_safe_2.address_a_1			unsigned short	255	$1* 10^0$ * (Raw+0)
			(none)	0	1

SNVT_safe_2.address_a_timedate	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 0)
SNVT_safe_2.address_a_2	bitfield	127	1* 10^0 * (Raw+0)
		0	1 (bits: 7) (offset: 1)
SNVT_safe_2.address_a_3	unsigned short	255	1* 10^0 * (Raw+0)
	(none)	0	1
SNVT_safe_2.time_stamp_msword	unsigned long	65 535	1* 10^0 * (Raw+0)
	(none)	0	1
SNVT_safe_2.data_a	unsigned short	255	1* 10^0 * (Raw+0)
	(none)	0	1
SNVT_safe_2.crc_a	unsigned short	255	1* 10^0 * (Raw+0)
	(none)	0	1
SNVT_safe_2.id_b_header	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 0)
SNVT_safe_2.id_b_format	bitfield	3	1* 10^0 * (Raw+0)
		0	1 (bits: 2) (offset: 1)
SNVT_safe_2.id_b_version	bitfield	3	1* 10^0 * (Raw+0)
		0	1 (bits: 2) (offset: 3)
SNVT_safe_2.id_b_length	bitfield	7	1* 10^0 * (Raw+0)
		0	1 (bits: 3) (offset: 5)
SNVT_safe_2.address_b_1	unsigned short	1	1* 10^0 * (Raw+0)
	(none)	0	1
SNVT_safe_2.address_b_timedate	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 0)
SNVT_safe_2.address_b_2	bitfield	127	1* 10^0 * (Raw+0)
		0	1 (bits: 7) (offset: 1)
SNVT_safe_2.address_b_3	unsigned short	255	1* 10^0 * (Raw+0)

	(none)	0	1
SNVT_safe_2.time_stamp_lsword	unsigned long	65 535	1* 10^0 * (Raw+0)
	(none)	0	1
SNVT_safe_2.data_b	unsigned short	255	1* 10^0 * (Raw+0)
	(none)	0	1
SNVT_safe_2.crc_b	unsigned short	255	1* 10^0 * (Raw+0)
	(none)	0	1

4.199 SNVT_safe_4

Overview: Safe protocol for 4 bytes data length.

Index	Size in Bytes	CNP Data Type			
207	22	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_safe_4.id_a_header			bitfield	1	1* 10^0 * (Raw+0)
				0	1 (bits: 1) (offset: 0)
SNVT_safe_4.id_a_format			bitfield	3	1* 10^0 * (Raw+0)
				0	1 (bits: 2) (offset: 1)
SNVT_safe_4.id_a_version			bitfield	3	1* 10^0 * (Raw+0)
				0	1 (bits: 2) (offset: 3)
SNVT_safe_4.id_a_length			bitfield	7	1* 10^0 * (Raw+0)
				0	1 (bits: 3) (offset: 5)
SNVT_safe_4.address_a_1			unsigned short	255	1* 10^0 * (Raw+0)
			(none)	0	1
SNVT_safe_4.address_a_timedate			bitfield	1	1* 10^0 * (Raw+0)
				0	1 (bits: 1) (offset: 0)
SNVT_safe_4.address_a_2			bitfield	127	1* 10^0 * (Raw+0)
				0	1 (bits: 7) (offset: 1)

SNVT_safe_4.address_a_3	unsigned short	255	1* 10^0 * (Raw+0)
	(none)	0	1
SNVT_safe_4.time_stamp_msword	unsigned long	65 535	1* 10^0 * (Raw+0)
	(none)	0	1
SNVT_safe_4.data_a	unsigned short	255	1* 10^0 * (Raw+0)
	(none)	0	1
SNVT_safe_4.crc_a	unsigned short	255	1* 10^0 * (Raw+0)
	(none)	0	1
SNVT_safe_4.id_b_header	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 0)
SNVT_safe_4.id_b_format	bitfield	3	1* 10^0 * (Raw+0)
		0	1 (bits: 2) (offset: 1)
SNVT_safe_4.id_b_version	bitfield	3	1* 10^0 * (Raw+0)
		0	1 (bits: 2) (offset: 3)
SNVT_safe_4.id_b_length	bitfield	7	1* 10^0 * (Raw+0)
		0	1 (bits: 3) (offset: 5)
SNVT_safe_4.address_b_1	unsigned short	255	1* 10^0 * (Raw+0)
	(none)	0	1
SNVT_safe_4.address_b_timedate	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 0)
SNVT_safe_4.address_b_2	bitfield	127	1* 10^0 * (Raw+0)
		0	1 (bits: 7) (offset: 1)
SNVT_safe_4.address_b_3	unsigned short	255	1* 10^0 * (Raw+0)
	(none)	0	1
SNVT_safe_4.time_stamp_lsword	unsigned long	65 535	1* 10^0 * (Raw+0)
	(none)	0	1
SNVT_safe_4.data_b	unsigned short	255	1* 10^0 * (Raw+0)

	(none)	0	1
SNVT_safe_4.crc_b	unsigned short	255	1* 10^0 * (Raw+0)
	(none)	0	1

4.200 SNVT_safe_8

Overview: Safe protocol for 8 bytes data length.

Index	Size in Bytes	CNP Data Type		
208	30	structure		
Field		CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_safe_8.id_a_header		bitfield	1	1* 10^0 * (Raw+0)
			0	1 (bits: 1) (offset: 0)
SNVT_safe_8.id_a_format		bitfield	3	1* 10^0 * (Raw+0)
			0	1 (bits: 2) (offset: 1)
SNVT_safe_8.id_a_version		bitfield	3	1* 10^0 * (Raw+0)
			0	1 (bits: 2) (offset: 3)
SNVT_safe_8.id_a_length		bitfield	7	1* 10^0 * (Raw+0)
			0	1 (bits: 3) (offset: 5)
SNVT_safe_8.address_a_1		unsigned short	255	1* 10^0 * (Raw+0)
		(none)	0	1
SNVT_safe_8.address_a_timedate		bitfield	1	1* 10^0 * (Raw+0)
			0	1 (bits: 1) (offset: 0)
SNVT_safe_8.address_a_2		bitfield	127	1* 10^0 * (Raw+0)
			0	1 (bits: 7) (offset: 1)
SNVT_safe_8.address_a_3		unsigned short	255	1* 10^0 * (Raw+0)
		(none)	0	1
SNVT_safe_8.time_stamp_msword		unsigned long	65 535	1* 10^0 * (Raw+0)
		(none)	0	1

SNVT_safe_8.data_a	unsigned short	255	1* 10^0 * (Raw+0)
	(none)	0	1
SNVT_safe_8.crc_a	unsigned short	255	1* 10^0 * (Raw+0)
	(none)	0	1
SNVT_safe_8.id_b_header	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 0)
SNVT_safe_8.id_b_format	bitfield	3	1* 10^0 * (Raw+0)
		0	1 (bits: 2) (offset: 1)
SNVT_safe_8.id_b_version	bitfield	3	1* 10^0 * (Raw+0)
		0	1 (bits: 2) (offset: 3)
SNVT_safe_8.id_b_length	bitfield	7	1* 10^0 * (Raw+0)
		0	1 (bits: 3) (offset: 5)
SNVT_safe_8.address_b_1	unsigned short	255	1* 10^0 * (Raw+0)
	(none)	0	1
SNVT_safe_8.address_b_timedate	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 0)
SNVT_safe_8.address_b_2	bitfield	127	1* 10^0 * (Raw+0)
		0	1 (bits: 7) (offset: 1)
SNVT_safe_8.address_b_3	unsigned short	255	1* 10^0 * (Raw+0)
	(none)	0	1
SNVT_safe_8.time_stamp_lsword	unsigned long	65 535	1* 10^0 * (Raw+0)
	(none)	0	1
SNVT_safe_8.data_b	unsigned short	255	1* 10^0 * (Raw+0)
	(none)	0	1
SNVT_safe_8.crc_b	unsigned short	255	1* 10^0 * (Raw+0)
	(none)	0	1

4.201 SNVT_time_val_2

Overview: Specifies the starting time and value for a scheduled event where the value of each event may consist of an occupancy value, a general purpose value, or both values

Index	Size	Data Type			
209	4 bytes	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SNVT_time_val_2.sunrise_relative_flag			bitfield	1	$1\times10^{0}\times(Raw+0)$
				0	bits: 1 at offset: 0
SNVT_time_val_2.sunset_relative_flag			bitfield	1	$1\times10^{0}\times(Raw+0)$
				0	bits: 1 at offset: 1
SNVT_time_val_2.negative_time_offset_flag			bitfield	1	$1\times10^{0}\times(Raw+0)$
				0	bits: 1 at offset: 2
SNVT_time_val_2.hour			bitfield	31	$1\times10^{0}\times(Raw+0)$
				0	bits: 5 at offset: 3
SNVT_time_val_2.occ_value_ignored_flag			bitfield	1	$1\times10^{0}\times(Raw+0)$
				0	bits: 1 at offset: 0
SNVT_time_val_2.gp_value_ignored_flag			bitfield	1	$1\times10^{0}\times(Raw+0)$
				0	bits: 1 at offset: 1
SNVT_time_val_2.minutes			bitfield	1	$1\times10^{0}\times(Raw+0)$
				0	bits: 6 at offset: 2
SNVT_time_val_2.occupancy			occup_t	3	none (enumeration)
			-1	-1	1
SNVT_time_val_2.gp_value			unsigned short	1	$1\times10^{0}\times(Raw+0)$
			(no invalid value)	0	1

4.202 SNVT_time_ offset

Overview: Time offset. Can be used to offset a scheduled start or stop time to implement an optimal start or optimal stop algorithm, or to stagger start times to minimize demand spikes.

Index	Size	Data Type	Minimum	Maximum	Scaling	Resolution
210	2 bytes	signed long	-32 768	32 766	$1\times10^{0}\times(Raw+0)$	1 s

4.203 SNVT_sched_exc

Overview: Specifies a scheduled event to override a daily schedule

Index	Size	Data Type			
211	8 bytes	structure			
Field		CNP Data Type / Invalid Value	Maximum / Minimum		Scaled Value / Resolution
SNVT_sched_exc.months		months_t	24		none (enumeration)
		-1	-1		1
SNVT_sched_exc.days		days_of_month_t	117		none (enumeration)
		-1	-1		1
SNVT_sched_exc.start_time_value		SNVT_time_val_2			
SNVT_sched_exc.end_time		structure			
SNVT_sched_exc.end_time.sunrise_relative_flag		bitfield	1		$1\times10^{0}\times(Raw+0)$
			0		bits: 1 at offset: 0
SNVT_sched_exc.end_time.sunset_relative_flag		bitfield	1		$1\times10^{0}\times(Raw+0)$
			0		bits: 1 at offset: 1
SNVT_sched_exc.end_time.negative_time_offset_flag		bitfield	1		$1\times10^{0}\times(Raw+0)$
			0		bits: 1 at offset: 2

SNVT_sched_exc.end_time.hour	bitfield	1	$1\times10^{0}\times(Raw+0)$
		0	bits: 5 at offset: 3
SNVT_sched_exc.end_time.start_offset_enable_flag	bitfield	1	$1\times10^{0}\times(Raw+0)$
		0	bits: 1 at offset: 0
SNVT_sched_exc.end_time.stop_offset_enable_flag	bitfield	1	$1\times10^{0}\times(Raw+0)$
		0	bits: 1 at offset: 1
SNVT_sched_exc.end_time.minutes	bitfield	1	$1\times10^{0}\times(Raw+0)$
		0	bits: 6 at offset: 2

4.204 SNVT_ sched_status

Overview: Type of schedule that is active

Index	Size	Data Type	Minimum	Maximum	Scaled Value	Invalid
212	1 bytes	scheduler_status_t	-1	6	none (enumeration)	-1

4.205 SNVT_ mass_flow

Overview: Rate of mass flow per unit of time

Index	Size	Data Type	Minimum	Maximum	Scaling	Resolution
213	2 bytes	signed long	-32768	32767	$1\times10^{-1}\times(Raw+0)$	0.1 kg/h

4.206 SNVT_ mass_flow_f

Overview: Rate of mass flow per unit of time

Index	Size	Data Type	Minimum	Maximum	Scaled Value	Invalid
214	4 bytes	floating-point	-3,402 823 466 385 3E + 038	3,402 823 466 385 3E + 038	floating	NAN and Out of Range

5 Standard configuration-property type – SCPT

5.1 Introduction

Most devices require customization for a specific system application. Configuration properties provide a standard mechanism that can be used by network tools to specify customization data for a CNP device. This document defines a standard set of configuration property types; these are called *standard configuration-property types* (SCPTs). Manufacturers may also define their own configuration property types; these are called *user configuration-property types* (UCPTs). SCPTs are defined for a wide range of configuration properties used in many kinds of functional profiles, such as hysteresis bands, default values, minimum and maximum limits, gain settings, and delay times. SCPTs are to be used wherever applicable. In situations where there is not an appropriate SCPT available, manufacturers may define UCPTs for configuring their devices, but these shall be documented in manufacturer-specific resource files. Each SNVT and SCPT is a scalar or structure type. A scalar type represents a single value that is a fixed-point number, floating-point number, or enumeration. A structure is a set of one or more scalar values, embedded structures, arrays, and/or unions. Each scalar type may define a minimum and maximum range, called the *valid range*. The limits for the valid range depend on the scalar type selected, as defined in EN 14908-5.

Configuration properties may be implemented as network variables, or may be implemented with data files called *configuration files*. Configuration files provide a compact mechanism for handling large amounts of configuration information on a device and do not use-up network-variable resources. Configuration files are transferred to or from a device via the CNP file-transfer protocol defined in 8.2.2.4, or by network-management read/write messages.

A list of all available SCPTs and details of their definitions is provided in this document. This document provides information on all available SCPTs. A SCPT index is defined for each configuration property that is used when defining self-documentation for configuration network variables, and also when defining configuration files. The SCPT names are provided for use in network and development tools.

5.2 SCPTactFbDly

Overview: Actual position feedback delay. The period for updating the feedback output when the actuator position does not match the requested position.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
1	7	SNVT_elapsed_tm				

5.3 SCPTalrmClrT1

Overview: Alarm clear time 1. The time period that an alarm 1 condition shall not exist before it is regarded as a valid cleared alarm.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
2	7	SNVT_elapsed_tm				

5.4 SCPTalrmClrT2

Overview: Alarm clear time 2. The time period that an alarm 2 condition shall not exist before it is regarded as a valid cleared alarm.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
3	7	SNVT_elapsed_tm				

5.5 SCPTalrmIhbT

Overview: Alarm output inhibit time. The time period for which alarms are inhibited after an object is enabled or the node is reset.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
4	7	SNVT_elapsed_tm				

5.6 SCPTalrmSetT1

Overview: Alarm set time 1. The time period that an alarm 1 condition shall exist before it is regarded as a valid alarm.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
5	7	SNVT_elapsed_tm				

5.7 SCPTalrmSetT2

Overview: Alarm set time 2. The time period that an alarm 2 condition shall exist before it is regarded as a valid alarm.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
6	7	SNVT_elapsed_tm				

5.8 SCPTdefOutput

Overview: Default output value.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
7		inheriting				

5.9 SCPTdriveT

Overview: Drive time. Time to be taken by the actuator to move from one extreme to the other.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
8	7	SNVT_elapsed_tm				

5.10 SCPThighLimit1

Overview: High-limit alarm 1.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
9		inheriting				

5.11 SCPThighLimit2

Overview: High-limit alarm 2.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
10		inheriting				

5.12 SCPThystHigh1

Overview: High-level hysteresis value 1.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
11		inheriting				

5.13 SCPThystHigh2

Overview: High-level hysteresis value 2.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
12		inheriting				

5.14 SCPThystLow1

Overview: Low-level hysteresis value 1.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
13		inheriting				

5.15 SCPThystLow2

Overview: Low-level hysteresis value 2.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
14		inheriting				

5.16 SCPTinFbDly

Overview: Input value feedback delay. The time period after the last update in a succession of changes to the input, before the feedback output is updated.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
15	7	SNVT_elapsed_tm				

5.17 SCPTinvrtOut

Overview: Invert output. This parameter indicates to invert the active polarity, if the value is nonzero (ON).

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
16	1	SNVT_lev_disc				

5.18 SCPTlocation

Overview: Location. Provides descriptive physical location information related to the object.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
17	31	SNVT_str_asc				

5.19 SCPTlowLimit1

Overview: Low-limit alarm 1.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
18		inheriting				

5.20 SCPTlowLimit2

Overview: Low-limit alarm 2.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
19		inheriting				

5.21 SCPTmaxRnge

Overview: Maximum range.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
20		inheriting				

5.22 SCPTmaxRcvT

Overview: Maximum receive time. The maximum time elapsed after the last update before the actuator adopts the default output

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
21	7	SNVT_elapsed_tm				

5.23 SCPTmaxSndT

Overview: Maximum send time. The maximum period of time between consecutive transmissions of the current value

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
22	7	SNVT_elapsed_tm				

5.24 SCPTminRnge

Overview: Minimum range.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
23		inheriting				

5.25 SCPTminSndT

Overview: Minimum send time. The minimum period of time between consecutive transmissions of the current value

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
24	7	SNVT_elapsed_tm				

5.26 SCPTnwrkCnfg

Overview: Network configuration source. The value of this field determines the source of the node's network configuration.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
25	1	SNVT_config_src				

5.27 SCPToffset

Overview: Offset. Typically used for calibration.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
26		inheriting				

5.28 SCPTsndDelta

Overview: Minimum delta for sending a update value.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
27		inheriting				

5.29 SCPTtrnsTblX

Overview: Translation table X. Used in conjunction with Translation table Y to scale and linearize a value

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
28	30	SNVT_trans_table				

5.30 SCPTtrnsTblY

Overview: Translation table Y. Used in conjunction with Translation table X to scale and linearize a value

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
29	30	SNVT_trans_table				

5.31 SCPToffDely

Overview: Turn-off delay. The length of time that the load remains energized after a change from ON to OFF has been received

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
30	7	SNVT_elapsed_tm				

5.32 SCPTgain

Overview: Gain. Typically used for calibration.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
31	4	SNVT_muldiv				

5.33 SCPTovrBehave

Overview: Override behavior. This parameter is used to define the behavior when an override request is received.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
32	1	SNVT_override				

5.34 SCPTovrValue

Overview: Override value.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
33		inheriting				

5.35 SCPTbypassTime

Overview: Bypass time. The maximum amount of time that the controller can be in the bypass (occupancy) mode following the last bypass request. Zero disables the timer.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
34	2	SNVT_time_min				

5.36 SCPTmanOvrTime

Overview: Manual override time. The maximum time that the controller will stay in a manual mode following the last request by a network variable input. Zero disables the timer.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
35	2	SNVT_time_min				

5.37 SCPThumSetpt

Overview: Humidity high limit setpoint. High limit humidity setpoint for the controlled space. A zero value disables

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
36	2	SNVT_lev_percent				

5.38 SCPTmaxFlowHeat

Overview: Maximum heating airflow. The maximum airflow setpoint of a VAV terminal while heating

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
37	2	SNVT_flow				

5.39 SCPTfireInitType

Overview: Fire initiator type identifier. The fire initiator type identifier, entered into the device at installation and/or configuration time

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
38	1	SNVT_fire_init				

5.40 SCPTsmokeNomSens

Overview: Nominal sensitivity. The nominal sensitivity value for the fire initiator in percentage obscuration by smoke

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
39	2	SNVT_smo_obscur				

5.41 SCPTsmokeDayAlrmLim

Overview: Daytime alarm limit. The daytime alarm limit sensitivity value for the fire initiator in percentage obscuration by smoke

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
40	2	SNVT_smo_obscur				

5.42 SCPTactuatorType

Overview: Actuator label. The identification of the exact actuator type or label

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
41	31	SNVT_str_asc				

5.43 SCPTlimitCO2

Overview: CO_2 limit. CO_2 threshold limit, controller to maintain concentration below this limit

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
42	2	SNVT_ppm				

5.44 SCPTminDeltaAngl

Overview: Damper angle send on delta. The minimum change in damper actuator angle required to be treated as significant

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
43	2	SNVT_angle_deg				

5.45 SCPTdirection

Overview: Direction / Safety position. The actuator sense of rotation and safety position; bit 0 set => counterclockwise, bit 1 set => damper open

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
44	2	SNVT_state				

5.46 SCPTdriveTime

Overview: Drive time. The transition time for a full 100 % stroke (change from one extreme to the other)

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
45	2	SNVT_time_sec				

5.47 SCPTductArea

Overview: Duct area or size. The duct area used to calculate the air flow, relevant only for VAV actuators / controllers

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
46	2	SNVT_area				

5.48 SCPTminDeltaFlow

Overview: Flow send on delta. The minimum change in airflow required to be treated as significant

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
47	2	SNVT_flow				

5.49 SCPTmaxRcvTime

Overview: Maximum receive time. The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
48	2	SNVT_time_sec				

5.50 SCPTmaxSendTime

Overview: Maximum send time. The maximum period of time between consecutive transmissions of the current value

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
49	2	SNVT_time_sec				

5.51 SCPTmaxSetpoint

Overview: Maximum setpoint. Either the maximum angle of rotation for an actuator or the maximum airflow for an actuator depending on actuator category

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
50	2	SNVT_lev_percent				

5.52 SCPTmaxFlow

Overview: Maximum flow.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
51	2	SNVT_flow				

5.53 SCPTminSendTime

Overview: Minimum send time. The minimum period of time between consecutive transmissions of the current value

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
52	2	SNVT_time_sec				

5.54 SCPTminSetpoint

Overview: Minimum setpoint. The minimum setpoint, such as minimum angle of rotation or minimum air flow

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
53	2	SNVT_lev_percent				

5.55 SCPTminFlow

Overview: Minimum flow.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
54	2	SNVT_flow				

5.56 SCPTminFlowHeat

Overview: Minimum heating airflow. The minimum airflow setpoint of a VAV terminal while heating

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
55	2	SNVT_flow				

5.57 SCPTminFlowStby

Overview: Minimum flow for standby. The minimum flow through the VAV box in standby mode

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
56	2	SNVT_flow				

5.58 SCPTnomAirFlow

Overview: Nominal air flow. Value used in calculating the air flow in an airflow control actuator

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
57	2	SNVT_flow				

5.59 SCPTnomAngle

Overview: Nominal angle. The nominal angle for an actuator

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
58	2	SNVT_angle_deg				

5.60 SCPTnumValves

Overview: Number of output valves. Used to inform the controller whether it is in a one-valve or two-valve system

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
59	2	SNVT_count				

5.61 SCPTsetPnts

Overview: Occupancy temperature setpoints. The occupancy temperature setpoints for heat and cool mode

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
60	12	SNVT_temp_setpt				

5.62 SCPToemType

Overview: OEM label. The label, programmed by the OEM, to identify the unit name

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
61	31	SNVT_str_asc				

5.63 SCPTminDeltaRH

Overview: Minimum delta relative humidity. The minimum change in RH level required to be treated as significant

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
62	2	SNVT_lev_percent				

5.64 SCPTminDeltaCO2

Overview: Minimum delta CO_2 level. The minimum change in CO_2 level required to be treated as significant

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
63	2	SNVT_ppm				

5.65 SCPTminDeltaTemp

Overview: Minimum delta temperature. The minimum change in temperature required to be treated as significant

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
64	2	SNVT_temp_p				

5.66 SCPTsensConstTmp

Overview: Temperature sensor constant. Calibration value for a duct temperature sensor

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
65	2	SNVT_multiplier				

5.67 SCPTgainVAV

Overview: VAV gain. The gain of the VAV controller object

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
66	2	SNVT_multiplier				

5.68 SCPTsensConstVAV

Overview: VAV sensor constant. Calibration constant used to calculate airflow

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
67	2	SNVT_multiplier				

5.69 SCPToffsetCO2

Overview: CO_2 level offset. Used to calibrate external hardware with additive offset after transformation

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
68	2	SNVT_ppm				

5.70 SCPToffsetRH

Overview: Relative humidity offset. Used to calibrate external hardware with additive offset after transformation

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
69	2	SNVT_lev_percent				

5.71 SCPToffsetTemp

Overview: Temperature offset. Used to calibrate external hardware with additive offset after transformation

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
70	2	SNVT_temp_p				

5.72 SCPTdefltBehave

Overview: Default behavior. Selects which set of values will be used on power-up and communication failure, between the stated default values (0), or manufacturer-specified values (1)

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
71	2	SNVT_switch				

5.73 SCPTpwrUpDelay

Overview: Power-up delay. The minimum period of time after power-up or re-establishment of communications before a control action takes place

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
72	2	SNVT_time_sec				

5.74 SCPTpwrUpState

Overview: Chiller enable. The default power-up and restart modes of the device when the default behavior selector is set to zero

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
73	2	SNVT_switch				

5.75 SCPThvacMode

Overview: HVAC mode. The default operating mode of the device when the default behavior selector is set to zero

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
74	1	SNVT_hvac_mode				

5.76 SCPTcoolSetpt

Overview: Cooling setpoint. The default setpoint for the leaving chilled water temperature in cooling mode when the default behavior selector is set to zero

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
75	2	SNVT_temp_p				

5.77 SCPTcoolLowerSP

Overview: Cooling setpoint lower limit. Limits the lower extent of the permitted range for the cooling setpoint

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
76	2	SNVT_temp_p				

5.78 SCPTcoolUpperSP

Overview: Cooling setpoint upper limit. Limits the upper extent of the permitted range for the cooling setpoint

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
77	2	SNVT_temp_p				

5.79 SCPTheatSetpt

Overview: Heating setpoint. The default setpoint for the leaving water temperature in heating mode when the default behavior selector is set to zero

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
78	2	SNVT_temp_p				

5.80 SCPTheatLowerSP

Overview: Heating setpoint lower limit. Limits the lower extent of the permitted range for the heating setpoint

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
79	2	SNVT_temp_p				

5.81 SCPTheatUpperSP

Overview: Heating setpoint upper limit. Limits the upper extent of the permitted range for the heating setpoint

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
80	2	SNVT_temp_p				

5.82 SCPTlimitChlrCap

Overview: Chiller capacity limit. The default value for the capacity limit of the chiller when the default behavior selector is set to zero

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
81	2	SNVT_lev_percent				

5.83 SCPTluxSetpoint

Overview: Setpoint, illumination level. The illumination level setpoint for the controller

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
82	2	SNVT_lux				

5.84 SCPTstep

Overview: Maximum step. The maximum step that the associated controller may take to approach the target level

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
83	1	SNVT_lev_cont				

5.85 SCPTonOffHysteresis

Overview: Hysteresis, auto mode on/off. Sets the hysteresis for the level setpoint. Setting to zero disables automatic on/off switching

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
84	1	SNVT_lev_cont				

5.86 SCPTclOffDelay

Overview: Controller off delay. The delay after which the controller output is switched off

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
85	2	SNVT_time_sec				

5.87 SCPTclOnDelay

Overview: Controller on delay. The delay after which the controller output is switched on

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
86	2	SNVT_time_sec				

5.88 SCPTpowerupState

Overview: Power-up state. The state of a light controller object after power-up or reset

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
87	4	SNVT_setting				

5.89 SCPTminDeltaLevel

Overview: Send on delta. The minimum change required to force transmission of the output value

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
88	1	SNVT_lev_cont				

5.90 SCPTreflection

Overview: Reflection factor. The internal gain factor for the measured illumination level

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
89	2	SNVT_lev_percent				

5.91 SCPTfieldCalib

Overview: Field calibration. Used by a light sensor to self calibrate the hardware

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
90	2	SNVT_lux				

5.92 SCPTholdTime

Overview: Hold time. Hold time for occupied state after there is no occupancy detected

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
91	2	SNVT_time_sec				

5.93 SCPTstepValue

Overview: Step value, ramp or master fade. The step value for up/down ramps or fade control

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
92	1	SNVT_lev_cont				

5.94 SCPTmaxOut

Overview: Maximum output value. The maximum value limit of the associated output network variable

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
93	1	SNVT_lev_cont				

5.95 SCPTsceneNmbr

Overview: Scene number in Numeric selector. The number of the first scene for the panel, other numbers are subsequent

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
94	1	unsigned short	1	255	$1* 10^{0} * (Raw+0)$	1 Numeric selector

5.96 SCPTfadeTime

Overview: Fade time. The desired time to fade to the scene value or to zero

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
95	2	SNVT_time_sec				

5.97 SCPTdelayTime

Overview: Delay time. The delay time prior to starting a fade to the scene value

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
96	2	SNVT_time_sec				

5.98 SCPTmasterSlave

Overview: Master-slave operation. Used to select master clock or slave clock, non-zero indicates this is master clock

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
97	1	unsigned short	0	1	$1* 10^{0} * (Raw+0)$	1 boolean

5.99 SCPTupdateRate

Overview: Update rate, time stamp. The update rate of the master clock to the associated network variable

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
98	2	SNVT_time_sec				

5.100 SCPTsummerTime

Overview: Summer time, start date and time. The start of summer time for purposes of daylight-savings time, all zeros disables

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
99	7	SNVT_time_stamp				

5.101 SCPTwinterTime

Overview: Winter time, start date and time. The start of winter time for purposes of daylight-savings time, all zeros disables

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
100	7	SNVT_time_stamp				

5.102 SCPTmanualAllowed

Overview: Manual allowed. Provides a clock, with a manual time input, the possibility to permit manual time updating

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
101	1	unsigned short	0	1	$1* 10^0 *$ (Raw+0)	1 boolean

5.103 SCPTdefWeekMask

Overview: Definition week mask. Day type definition for every day of the week

Index	Size in Bytes	CNP Data Type			
102	14	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SCPTdefWeekMask.time_event_index			unsigned long	65 535	$1* 10^0 *$ (Raw+0)
			(none)	0	1 array of 7 index values

5.104 SCPTdayDateIndex

Overview: Day date index. One or two dates for matching with a start index to the time-event array

Index	Size in Bytes	CNP Data Type			
103	6	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SCPTdayDateIndex.day_1			unsigned short	31	1* 10^0 * (Raw+0)
			0	1	1 day
SCPTdayDateIndex.month_1			unsigned short	12	1* 10^0 * (Raw+0)
			0	1	1 month
SCPTdayDateIndex.day_2			unsigned short	31	1* 10^0 * (Raw+0)
			0	0	1 day
SCPTdayDateIndex.month_2			unsigned short	12	1* 10^0 * (Raw+0)
			0	0	1 month
SCPTdayDateIndex.event_mode_index			unsigned long	65 535	1* 10^0 * (Raw+0)
			0	0	1 index

5.105 SCPTtimeEvent

Overview: Time event entry. Event or mode definitions to be transmitted if the time in the record is reached

Index	Size in Bytes	CNP Data Type			
104	4	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SCPTtimeEvent.record_type			event_mode_type_t	4	none (enumeration)
			-1	-1	1
SCPTtimeEvent.hour			unsigned short	23	$1* 10^0 * (Raw+0)$
			(none)	0	1 h
SCPTtimeEvent.minute			unsigned short	59	$1* 10^0 * (Raw+0)$
			(none)	0	1 min
SCPTtimeEvent.event_mode			unsigned short	255	$1* 10^0 * (Raw+0)$
			(none)	0	1 event mode number

5.106 SCPTmodeHrtBt

Overview: Heart beat, mode output. The time that shall pass without an update for mode definitions to be automatically retransmitted, zero disables

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
105	2	SNVT_time_sec				

5.107 SCPTdefrostMode

Overview: Defrost mode. The type of defrost to perform

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
106	1	SNVT_defr_mode				

5.108 SCPTmaxDefrstTime

Overview: Maximum defrost time. The maximum defrost time for defrost objects set to terminate on temperature

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
107	2	SNVT_time_sec				

5.109 SCPTdrainDelay

Overview: Drain delay. The delay to use after the defrost has terminated

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
108	2	SNVT_time_sec				

5.110 SCPTinjDelay

Overview: Injection delay. The delay to use after the defrost has terminated

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
109	2	SNVT_time_sec				

5.111 SCPTmaxDefrstTemp

Overview: Defrost stop temperature. The temperature at which to terminate defrost for objects set to terminate on temperature

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
110	2	SNVT_temp_p				

5.112 SCPTstrtupDelay

Overview: Startup delay. The time to delay after power-up, defrost, or pack fail

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
111	2	SNVT_time_sec				

5.113 SCPTtermTimeTemp

Overview: Defrost termination setting. The defrost termination condition

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
112	1	SNVT_defr_term				

5.114 SCPTpumpDownDelay

Overview: Pump down delay. The delay to use before starting the defrost

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
113	2	SNVT_time_sec				

5.115 SCPTsuperHtRefInit

Overview: Super heat reference initialization. Default value for the super heat target network variable

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
114	2	SNVT_temp_p				

5.116 SCPTstrtupOpen

Overview: Startup valve opening. Maximum valve opening to use after power-up, defrost, or pack fail

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
115	2	SNVT_lev_percent				

5.117 SCPTsuperHtRefMin

Overview: Super heat reference minimum. Minimum value for the target super heat network variable

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
116	2	SNVT_temp_p				

5.118 SCPTrefrigGlide

Overview: Refrigerant glide. Used to characterize the glide of the refrigerant used

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
117	2	SNVT_temp				

5.119 SCPTsuperHtRefMax

Overview: Super heat reference maximum. Maximum value for the target super heat network variable

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
118	2	SNVT_temp_p				

5.120 SCPTrefrigType

Overview: Refrigerant type.

Index	Size in Bytes	CNP Data Type			
119	18	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SCPTrefrigType.refrigerant			unsigned char	255	$1* 10^{0} * (Raw+0)$
			(none)	0	1 array of 6 characters
SCPTrefrigType.A			floating-point	3,402 823 466 385 3E + 038	floating
			NAN and Out of Range	-3,402 823 466 385 3E + 038	Constant A
SCPTrefrigType.B			floating-point	3,402 823 466 385 3E + 038	floating
			NAN and Out of Range	-3,402 823 466 385 3E + 038	Constant B
SCPTrefrigType.C			floating-point	3,402 823 466 385 3E + 038	floating
			NAN and Out of Range	-3,402 823 466 385 3E + 038	Constant C

5.121 SCPTthermMode

Overview: Thermostat mode. The thermostat control strategy

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
120	1	SNVT_therm_mode				

5.122 SCPTdayNightCntrl

Overview: Day/night control. Configures the day/night function

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
121	2	SNVT_state				

5.123 SCPTdiffNight

Overview: Difference night. The value to be added to the cut-out value to get the cut-in limit during night control

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
122	2	SNVT_temp_p				

5.124 SCPThighLimTemp

Overview: High-limit temperature. The high alarm set point for the alarm air temp network variable

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
123	2	SNVT_temp_p				

5.125 SCPThighLimDly

Overview: High-limit delay. The time limit during normal operation before the alarm air temp high alarm is recognized

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
124	2	SNVT_time_sec				

5.126 SCPTcutOutValue

Overview: Cut-out value. The cut-out limit

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
125	2	SNVT_temp_p				

5.127 SCPTairTemp1Day

Overview: Air temperature 1 percent day. The air temperature weighting used during day control

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
126	2	SNVT_lev_percent				

5.128 SCPTsmokeNightAlrmLim

Overview: Nighttime alarm limit. The nighttime alarm limit sensitivity value for the fire initiator in percentage obscuration by smoke

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
127	2	SNVT_smo_obscur				

5.129 SCPTlowLimTemp

Overview: Low-limit temperature. The low alarm set point for the alarm air temperature network variable

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
128	2	SNVT_temp_p				

5.130 SCPTlowLimDly

Overview: Low-limit delay. The time limit during normal operation before the alarm air temperature low alarm is recognized

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
129	2	SNVT_time_sec				

5.131 SCPTdiffValue

Overview: Difference value. The value to be added to the cut-out value to get the cut-in limit

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
130	2	SNVT_temp_p				

5.132 SCPTairTemp1Night

Overview: Air temperature 1 percent night. The air temperature weighting used during night control

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
131	2	SNVT_lev_percent				

5.133 SCPTairTemp1Alrm

Overview: Air temperature 1 percent alarm. The weighting of the air temperature 1 sensor when calculating the air temperature alarm

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
132	2	SNVT_lev_percent				

5.134 SCPThighLimDefrDly

Overview: High-limit defrost delay. The time limit before high air temperature alarm during pull-down

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
133	2	SNVT_time_sec				

5.135 SCPTdeltaNight

Overview: Delta night. The value to be added to the cut-out value to get the cut-out limit during night control

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
134	2	SNVT_temp_p				

5.136 SCPTrunHrInit

Overview: Running hours counter initialization. The initial value of the running hours counter network variable

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
135	7	SNVT_elapsed_tm				

5.137 SCPTrunHrAlarm

Overview: Running hours alarm threshold level. The alarm threshold for the running hours counter

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
136	7	SNVT_elapsed_tm				

5.138 SCPTenergyCntInit

Overview: Energy counter initialization. The initial value of the energy counter for the associated network variable

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
137	2	SNVT_elec_kwh				

5.139 SCPTsmokeDayPreAlrmLim

Overview: Daytime pre-alarm limit. The daytime pre-alarm limit sensitivity value for the fire initiator in percentage obscuration by smoke

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
138	2	SNVT_smo_obscur				

5.140 SCPTdebounce

Overview: Debounce time. The debouncing time to generate the detection envelope

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
139	2	SNVT_time_sec				

5.141 SCPTsmokeNightPreAlrmLim

Overview: Nighttime pre-alarm limit. The nighttime pre-alarm limit sensitivity value for the fire initiator in percentage obscuration by smoke

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
140	2	SNVT_smo_obscur				

5.142 SCPTzoneNum

Overview: Zone number in zone number. The zone number for the device

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
141	2	unsigned long	0	65 535	$1* 10^{0} * (Raw+0)$	1 zone number

5.143 SCPTthermAlrmROR

Overview: Thermal rate of change/rise trip value. The thermal alarm trip rate of rise

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
142	2	SNVT_temp_ror				

5.144 SCPTvisOutput

Overview: Visible light output intensity in candela. Visible light output intensity specification of the device at 0 degree viewing angle

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
143	2	unsigned long	0	10 000	$1* 10^{-1} * (Raw+0)$	0,1 candela

5.145 SCPTaudOutput

Overview: Audible sound output intensity. Audible sound output intensity specification of the device at 1 m distance

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
144	2	SNVT_sound_db				

5.146 SCPTflashFreq

Overview: Flash rate specification. Flash rate specification for visible indication (strobe) device

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
145	2	SNVT_freq_hz				

5.147 SCPTinstallDate

Overview: Installation date. The date of installation for the device

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
146	7	SNVT_time_stamp				

5.148 SCPTmaintDate

Overview: Maintenance date. The date of last maintenance for the device

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
147	7	SNVT_time_stamp				

5.149 SCPTmanfDate

Overview: Manufacture date. The date of manufacture for the device

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
148	7	SNVT_time_stamp				

5.150 SCPTfireTxt1

Overview: Fire text information. Text information relevant to fire conditions. A '>' at end of string indicates presence of fire text 2.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
149	31	SNVT_str_asc				

5.151 SCPTfireTxt2

Overview: Fire text information, continuation. Continuation text information relevant to fire conditions. A '>' at end of string indicates presence of fire text 3.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
150	31	SNVT_str_asc				

5.152 SCPTfireTxt3

Overview: Fire text information, second continuation. Second continuation text information relevant to fire conditions

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
151	31	SNVT_str_asc				

5.153 SCPTthermThreshold

Overview: Thermal alarm trip threshold. Thermal alarm trip threshold for the initiator

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
152	2	SNVT_temp				

5.154 SCPTfireIndicate

Overview: Fire indicator device type. Describes the fire indicator device

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
153	1	SNVT_fire_indcte				

5.155 SCPTtimeZone

Overview: Time zone descriptor. Time zone of node (offset from GMT, start and end of DST)

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
154	15	SNVT_time_zone				

5.156 SCPTprimeVal

Overview: Primary default value. The default output value when an area is occupied

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
155	2	SNVT_switch				

5.157 SCPTsecondVal

Overview: Secondary default value. The default output value when the neighboring area is occupied

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
156	2	SNVT_switch				

5.158 SCPTsceneOffset

Overview: Scene offset. The offset for the scene number when data is forwarded from primary to secondary

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
157	1	unsigned short	0	255	$1* 10^0 *$ (Raw+0)	1

5.159 SCPTnomRPM

Overview: Nominal motor speed. The nominal speed of a motor in RPM

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
158	2	SNVT_rpm				

5.160 SCPTnomFreq

Overview: Nominal motor frequency. The nominal frequency of a motor

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
159	2	SNVT_freq_hz				

5.161 SCPTrampUpTm

Overview: Minimum ramp-up time. The ramp-up time of the device

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
160	2	SNVT_time_sec				

5.162 SCPTrampDownTm

Overview: Minimum ramp-down time. The ramp-down time of the device

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
161	2	SNVT_time_sec				

5.163 SCPTdefScale

Overview: Default speed scale. Default value for the speed scale

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
162	2	SNVT_lev_percent				

5.164 SCPTregName

Overview: Register name. The name of a utility data logger register device

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
163	31	SNVT_str_asc				

5.165 SCPTbaseValue

Overview: Base value. The base value (where to begin counting)

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
164	6	SNVT_reg_val				

5.166 SCPTdevMajVer

Overview: Device major version number. The major version number for the device

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
165	1	unsigned short	0	255	$1* 10^{0} * (Raw+0)$	1

5.167 SCPTdevMinVer

Overview: Device minor version number. The minor version number for the device

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
166	1	unsigned short	0	255	$1* 10^{0} * (Raw+0)$	1

5.168 SCPTobjMajVer

Overview: Object major version number. The major version number for the object

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
167	1	unsigned short	0	255	$1* 10^{0} * (Raw+0)$	1

5.169 SCPTobjMinVer

Overview: Object minor version number. The minor version number for the object

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
168	1	unsigned short	0	255	$1* 10^{0} * (Raw+0)$	1

5.170 SCPThvacType

Overview: HVAC unit type. The type of HVAC equipment being controlled

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
169	1	SNVT_hvac_type				

5.171 SCPTtimeout

Overview: Response timeout. The timeout for a controlling device to respond, during control permission request

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
170	2	SNVT_time_sec				

5.172 SCPTcontrolPriority

Overview: Control priority. Priority of a control input or output, lower values mean higher priority

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
171	1	unsigned short	0	200	$1* 10^{0} * (Raw+0)$	1 priority value

5.173 SCPTdeviceGroupID

Overview: Group ID. A logical group control ID for the device

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
172	2	unsigned long	0	65 535	$1* 10^{0} * (Raw+0)$	1 ID number

5.174 SCPTmaxPrivacyZones

Overview: Maximum privacy zones in units. The maximum number of privacy zones supported by a device

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
173	1	unsigned short	0	255	$1* 10^{0} * (Raw+0)$	1 units

5.175 SCPTmaxCameraPrepositions

Overview: Maximum pre-positions in units. The maximum number of pre-positions supported by a device

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
174	1	unsigned short	0	255	$1* 10^{0} * (Raw+0)$	1 units

5.176 SCPTdefaultPanTiltZoomSpeeds

Overview: Default camera PTZ. The default camera pan, tilt, and zoom speeds

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
175	6	SNVT_ptz				

5.177 SCPTdefaultAutoPanSpeed

Overview: Default auto-pan speed in percentage of full level. The default auto-pan speed for a device

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
176	1	unsigned short	0	250	$4 * 10^{-1} * (Raw+0)$	0,4 % of full level

5.178 SCPTautoAnswer

Overview: Auto answer. Enables the automatic call answer function of a device

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
177	1	boolean_t	-1	1	none (enumeration)	-1

5.179 SCPTdialString

Overview: Dial string. Telephone number string used in dialing, including characters used for control

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
178	31	SNVT_str_asc				

5.180 SCPTserialNumber

Overview: Serial number. Manufacturer-defined serial number string for the device

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
179	31	SNVT_str_asc				

5.181 SCPTnormalRotationalSpeed

Overview: Normal rotational speed. The normal rotational speed in Hz

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
180	2	SNVT_freq_hz				

5.182 SCPTstandbyRotationalSpeed

Overview: Standby rotational speed. The standby rotational speed in Hz

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
181	2	SNVT_freq_hz				

5.183 SCPTpartNumber

Overview: Part number. Manufacturer-defined part number string for the device

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
182	31	SNVT_str_asc				

5.184 SCPTdischargeAirCoolingSetpoint

Overview: Discharge air cooling setpoint. Default cooling setpoint for discharge air

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
183	2	SNVT_temp_p				

5.185 SCPTdischargeAirHeatingSetpoint

Overview: Discharge air heating setpoint. Default heating setpoint for discharge air

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
184	2	SNVT_temp_p				

5.186 SCPTmaxSupplyFanCapacity

Overview: Maximum supply fan capacity. Setpoint for maximum supply fan capacity

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
185	2	SNVT_lev_percent				

5.187 SCPTminSupplyFanCapacity

Overview: Minimum supply fan capacity. Setpoint for minimum supply fan capacity

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
186	2	SNVT_lev_percent				

5.188 SCPTmaxReturnExhaustFanCapacity

Overview: Maximum return/exhaust fan capacity. Setpoint for maximum return/exhaust fan capacity

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
187	2	SNVT_lev_percent				

5.189 SCPTminReturnExhaustFanCapacity

Overview: Minimum return/exhaust fan capacity. Setpoint for minimum return/exhaust fan capacity

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
188	2	SNVT_lev_percent				

5.190 SCPTductStaticPressureSetpoint

Overview: Duct static pressure. Setpoint for default duct static pressure

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
189	2	SNVT_press_p				

5.191 SCPTmaxDuctStaticPressureSetpoint

Overview: Maximum duct static pressure. Setpoint for maximum duct static pressure

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
190	2	SNVT_press_p				

5.192 SCPTminDuctStaticPressureSetpoint

Overview: Minimum duct static pressure. Setpoint for minimum duct static pressure

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
191	2	SNVT_press_p				

5.193 SCPTductStaticPressureLimit

Overview: Duct static pressure limit. The duct static pressure limit for equipment protection

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
192	2	SNVT_press_p				

5.194 SCPTbuildingStaticPressureSetpoint

Overview: Building static pressure. Setpoint for the default static pressure for the building

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
193	2	SNVT_press_p				

5.195 SCPTreturnFanStaticPressureSetpoint

Overview: Return fan pressure. Setpoint for the return fan static pressure

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
194	2	SNVT_press_p				

5.196 SCPTfanDifferentialSetpoint

Overview: Fan differential. Setpoint for the percent capacity differential between the supply and return fans

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
195	2	SNVT_lev_percent				

5.197 SCPTmixedAirLowLimitSetpoint

Overview: Mixed air low limit. Setpoint for the mixed air low temperature limit

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
196	2	SNVT_temp_p				

5.198 SCPTmixedAirTempSetpoint

Overview: Mixed air temperature. Setpoint for the default mixed air temperature

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
197	2	SNVT_temp_p				

5.199 SCPTminOutdoorAirFlowSetpoint

Overview: Minimum outdoor air flow. Setpoint for the default minimum outdoor air flow

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
198	2	SNVT_flow				

5.200 SCPToutdoorAirTempSetpoint

Overview: Outdoor air temperature. Setpoint for the outdoor air temperature economizer enable

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
199	2	SNVT_temp_p				

5.201 SCPToutdoorAirEnthalpySetpoint

Overview: Outdoor air enthalpy. Setpoint for the outdoor air enthalpy economizer enable

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
200	2	SNVT_enthalpy				

5.202 SCPTdiffTempSetpoint

Overview: Differential temperature. Setpoint for differential temperature for economizer enable

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
201	2	SNVT_temp_diff_p				

5.203 SCPTexhaustEnablePosition

Overview: Exhaust enable position. Setpoint for the exhaust-enable outdoor air damper position

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
202	2	SNVT_lev_percent				

5.204 SCPTspaceHumSetpoint

Overview: Space humidification. Setpoint for the default space humidification

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
203	2	SNVT_lev_percent				

5.205 SCPTdischargeAirDewpointSetpoint

Overview: Discharge air dewpoint. Setpoint for the default discharge air dewpoint

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
204	2	SNVT_temp_p				

5.206 SCPTmaxDischargeAirCoolingSetpoint

Overview: Maximum discharge air cooling. Setpoint for the maximum discharge air cooling

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
205	2	SNVT_temp_p				

5.207 SCPTminDischargeAirCoolingSetpoint

Overview: Minimum discharge air cooling. Setpoint for the minimum discharge air cooling

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
206	2	SNVT_temp_p				

5.208 SCPTmaxDischargeAirHeatingSetpoint

Overview: Maximum discharge air heating. Setpoint for the maximum discharge air heating

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
207	2	SNVT_temp_p				

5.209 SCPTminDischargeAirHeatingSetpoint

Overview: Minimum discharge air heating. Setpoint for the minimum discharge air heating

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
208	2	SNVT_temp_p				

5.210 SCPTcoolingLockout

Overview: Cooling lockout. Setpoint for the outdoor air temperature at which cooling will be disabled

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
209	2	SNVT_temp_p				

5.211 SCPTheatingLockout

Overview: Heating lockout. Setpoint for the outdoor air temperature at which heating will be disabled

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
210	2	SNVT_temp_p				

5.212 SCPTcoolingResetEnable

Overview: Cooling reset enable. The cooling reset control is enabled

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
211	1	boolean_t	-1	1	none (enumeration)	-1

5.213 SCPTheatingResetEnable

Overview: Heating reset enable. The heating reset control is enabled

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
212	1	boolean_t	-1	1	none (enumeration)	-1

5.214 SCPTsetpoint

Overview: Heating/cooling setpoint.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
213		inheriting				

5.215 SCPTtemperatureHysteresis

Overview: Temperature hysteresis. General-purpose temperature hysteresis differential between on-point and off-point

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
214	2	SNVT_temp_diff_p				

5.216 SCPTcontrolTemperatureWeighting

Overview: Temperature weighting. Weighting between two temperatures in percent

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
215	2	SNVT_lev_percent				

5.217 SCPTpwmPeriod

Overview: Pulse-width modulation period. The time period to be used in pulse-width modulation control strategy

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
216	7	SNVT_elapsed_tm				

5.218 SCPTdefrostInternalSchedule

Overview: Defrost internal schedule. Enable the internal scheduling of defrost

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
217	1	boolean_t	-1	1	none (enumeration)	-1

5.219 SCPTdefrostStart

Overview: Defrost start time. Start time for the first daily defrost

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
218	7	SNVT_time_stamp				

5.220 SCPTdefrostCycles

Overview: Defrost cycles in units. Number of equally-spaced defrost cycles to perform per day

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
219	1	unsigned short	0	255	$1* 10^{0} * (Raw+0)$	1 units

5.221 SCPTminDefrostTime

Overview: Minimum defrost time. Minimum time for defrost to run if terminated on temperature

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
220	2	SNVT_time_min				

5.222 SCPTmaxDefrostTime

Overview: Maximum defrost time. Maximum time for defrost to run if terminated on temperature

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
221	2	SNVT_time_min				

5.223 SCPTdefrostFanDelay

Overview: Fan delay after defrost. Delay after refrigeration is resumed after defrost before fan is started

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
222	2	SNVT_time_min				

5.224 SCPTdefrostRecoveryTime

Overview: Defrost recovery time. Time allowed after defrost finishes for temperature to be within normal limits

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
223	2	SNVT_time_min				

5.225 SCPTdefrostHold

Overview: Defrost hold on sync. Enables hold in defrost mode until synchronization signal goes away

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
224	1	boolean_t	-1	1	none (enumeration)	-1

5.226 SCPTdefrostDetect

Overview: Defrost detect temperature differential. Temperature differential of coil refrigerant temperature above discharge air temperature that indicates defrost

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
225	2	SNVT_temp_diff_p				

5.227 SCPTscheduleInternal

Overview: Internal schedule. Enables internal scheduling

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
226	1	boolean_t	-1	1	none (enumeration)	-1

5.228 SCPTtempOffset

Overview: Temperature offset. Temperature offset for calibration

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
227	2	SNVT_temp_diff_p				

5.229 SCPTaudibleLevel

Overview: Audible level. The audible level output of the device

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
228	2	SNVT_switch				

5.230 SCPTscrollSpeed

Overview: Scroll speed. The scroll speed of the display image

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
229	2	SNVT_switch				

5.231 SCPTbrightness

Overview: Brightness output. The brightness output of a display device

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
230	2	SNVT_switch				

5.232 SCPTorientation

Overview: Orientation. The orientation angle of the display image (0 = landscape, 90 = portrait)

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
231	2	SNVT_angle_deg				

5.233 SCPTinstalledLevel

Overview: Installed level in floor number. The floor or level on which the device is installed

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
232	2	unsigned long	0	65535	$1* 10^{0} * (Raw+0)$	1 floor number

5.234 SCPTpumpCharacteristic

Overview: Pump characteristic. The basic characteristic data for a pump

Index	Size in Bytes	CNP Data Type			
233	6	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SCPTpumpCharacteristic.speedMax			SNVT_rpm		
SCPTpumpCharacteristic.pressMax			SNVT_press		
SCPTpumpCharacteristic.flowMax			SNVT_flow_p		

5.235 SCPTminPressureSetpoint

Overview: Minimum pressure. Setpoint for the operational low pressure limit

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
234	2	SNVT_press				

5.236 SCPTmaxPressureSetpoint

Overview: Maximum pressure. Setpoint for the operational high pressure limit

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
235	2	SNVT_press				

5.237 SCPTminFlowSetpoint

Overview: Minimum flow. Setpoint for the operational low flow limit

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
236	2	SNVT_flow_p				

5.238 SCPTmaxFlowSetpoint

Overview: Maximum flow. Setpoint for the operational high flow limit

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
237	2	SNVT_flow_p				

5.239 SCPTdeviceControlMode

Overview: Device control mode. Normal default operating device control mode

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
238	1	SNVT_dev_c_mode				

5.240 SCPTminRemotePressureSetpoint

Overview: Minimum remote pressure. Setpoint for the operational low pressure limit (remote sensor)

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
239	2	SNVT_press				

5.241 SCPTmaxRemotePressureSetpoint

Overview: Maximum remote pressure. Setpoint for the operational high pressure limit (remote sensor)

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
240	2	SNVT_press				

5.242 SCPTminRemoteFlowSetpoint

Overview: Minimum remote flow. Setpoint for the operational low flow limit (remote sensor)

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
241	2	SNVT_flow_p				

5.243 SCPTmaxRemoteFlowSetpoint

Overview: Maximum remote flow. Setpoint for the operational high flow limit (remote sensor)

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
242	2	SNVT_flow_p				

5.244 SCPTminRemoteTempSetpoint

Overview: Minimum remote temperature. Setpoint for the operational low temperature limit (remote sensor)

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
243	2	SNVT_temp_p				

5.245 SCPTmaxRemoteTempSetpoint

Overview: Maximum remote temperature. Setpoint for the operational high temperature limit (remote sensor)

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
244	2	SNVT_temp_p				

5.246 SCPTcontrolSignal

Overview: Control signal. Start and end points (X,Y) for a transition

Index	Size in Bytes	CNP Data Type			
245	8	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SCPTcontrolSignal.x1Value			SNVT_lev_percent		
SCPTcontrolSignal.y1Value			SNVT_lev_percent		
SCPTcontrolSignal.x2Value			SNVT_lev_percent		
SCPTcontrolSignal.y2Value			SNVT_lev_percent		

5.247 SCPTnightPurgePosition

Overview: Night purge valve position. Valve position in percent open for night purge

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
246	2	SNVT_lev_percent				

5.248 SCPTfreeCoolPosition

Overview: Free cooling valve position. Valve position in percent open for free cooling HVAC mode

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
247	2	SNVT_lev_percent				

5.249 SCPTvalveFlowCharacteristic

Overview: Valve flow characteristic. Actual flow characteristic of the valve

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
248	1	SNVT_dev_c_mode				

5.250 SCPTvalveOperatingMode

Overview: Valve operating mode. The normal operating mode of the valve

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
249	1	SNVT_valve_mode				

5.251 SCPTemergencyPosition

Overview: Emergency position. Position in percent of full scale (open) for emergency operation

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
250	2	SNVT_lev_percent				

5.252 SCPTblockProtectionTime

Overview: Minimum time for movement. The minimum time in hours for movement to prevent blocking

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
251	2	SNVT_time_hour				

5.253 SCPTminStroke

Overview: Minimum stroke. The minimum stroke limit

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
252	2	SNVT_length_mil				

5.254 SCPTmaxStroke

Overview: Maximum stroke. The maximum stroke limit

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
253	2	SNVT_length_mil				

5.255 SCPTnvType

Overview: Network variable type. Network variable type for network variables that support changeable types

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
254	19	SNVT_nv_type				

5.256 SCPTmaxNVLength

Overview: Maximum network variable length in bytes. Maximum length of a type that may be assigned to the network variable

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
255	1	unsigned short	1	31	$1* 10^0 * (Raw+0)$	1 bytes

5.257 SCPTnvDynamicAssignment

Overview: Network variable dynamic assignment. Assigns a dynamic network variable to a functional block member

Index	Size in Bytes	CNP Data Type			
256	25	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SCPTnvDynamicAssignment.nv_index			unsigned long	4095	$1* 10^0 * (Raw+0)$
			65 535	0	1 nv index
SCPTnvDynamicAssignment.fblock_index			unsigned long	4 095	$1* 10^0 * (Raw+0)$
			65 535	0	1 fblock index
SCPTnvDynamicAssignment.member_number			unsigned long	4 095	$1* 10^0 * (Raw+0)$
			(none)	1	1 nv member number
SCPTnvDynamicAssignment.nv_type			SNVT_nv_type		

5.258 SCPTsafExtCnfg

Overview: Safety mode. Mode that a device has to be brought to when a "safety external request" state is pending.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
257		inheriting				

5.259 SCPTemergCnfg

Overview: Emergency mode. Mode that a device has to be brought to when an "emergency request" state is pending.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
258		inheriting				

5.260 SCPTsluiceCnfg

Overview: Sluice-lock master/slave control. Role of a device in a sluice-lock connection. A sluice-lock is an interlock mechanism between two entry/exit devices, or a sluice manager and several entry/exit devices, to ensure that only one single entry/exit device is opened at any point in time.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
259	1	master_slave_t	-1	2	none (enumeration)	-1

5.261 SCPTfanOperation

Overview: Fan operation.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
260	1	fan_operation_t	-1	4	none (enumeration)	-1

5.262 SCPTminFlowUnit

Overview: Unit minimum air flow. Unit minimum air flow for dual duct VAV Terminal units

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
261	2	SNVT_flow				

5.263 SCPTmaxFlowUnit

Overview: Unit maximum air flow. Unit maximum airflow for dual duct VAV Terminal units

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
262	2	SNVT_flow				

5.264 SCPTminFlowHeatStby

Overview: Standby heating minimum air flow. Heating or ventilated deck minimum flow of a dual duct VAV Terminal unit during occupied standby mode

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
263	2	SNVT_flow				

5.265 SCPTminFlowUnitStby

Overview: Standby unit minimum air flow. Total unit minimum airflow for dual duct units during occupied standby mode

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
264	2	SNVT_flow				

5.266 SCPToffsetFlow

Overview: Air flow offset. Offset value used to calculate the active airflow setpoint by adding nciFlowOffset to nviAirFlowSetpt

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
265	4	SNVT_flow_f				

5.267 SCPTareaDuctHeat

Overview: Heating duct area. Nominal cross-sectional airflow area of the hot or ventilation deck of a dual duct VAV terminal unit

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
266	2	SNVT_area				

5.268 SCPTnomAirFlowHeat

Overview: Heating nominal flow. Nominal airflow volume of a hot or ventilation deck of a dual duct VAV terminal

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
267	2	SNVT_flow				

5.269 SCPTgainVAVHeat

Overview: VAV sensor constant. Calibration constant used to calculate airflow

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
268	2	SNVT_multiplier				

5.270 SCPTnumDampers

Overview: Number of dampers. Used to inform the controller if it is in a single or dual duct system

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
269	2	SNVT_count				

5.271 SCPTminFlowUnitHeat

Overview: Unit Heating Minimum Flow. Minimum airflow setpoint of a single duct, or the unit minimum airflow setpoint of a dual duct VAV terminal when using a unit (local) heating source

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
270	2	SNVT_flow				

5.272 SCPTsaturationDelay

Overview: Saturation delay.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
271	2	SNVT_time_min				

5.273 SCPTeffectivePeriod

Overview: Effective period. Time period during which a functional block is effective.

Index	Size in Bytes	CNP Data Type			
272	8	structure			
Field		CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution	
SCPTeffectivePeriod.start		structure			
		Starting date of the effective period			
SCPTeffectivePeriod.start.year		signed long	3 000	$1* 10^0 * (Raw+0)$	
		-1	-1	1	
SCPTeffectivePeriod.start.month		unsigned short	12	$1* 10^0 * (Raw+0)$	
		(none)	1	1	
SCPTeffectivePeriod.start.day		unsigned short	31	$1* 10^0 * (Raw+0)$	
		(none)	1	1	
SCPTeffectivePeriod.end		structure			
		Ending date of the effective period			
SCPTeffectivePeriod.end.year		signed long	3 000	$1* 10^0 * (Raw+0)$	
		-1	-1	1	

SCPTeffectivePeriod.end.month	unsigned short	12	1* 10^0 * (Raw+0)
	(none)	1	1
SCPTeffectivePeriod.end.day	unsigned short	31	1* 10^0 * (Raw+0)
	(none)	1	1

5.274 SCPTscheduleDates

Overview: Schedule dates. A range of dates with an optional qualifier that specifies when a schedule is active

Index	Size in Bytes	CNP Data Type			
273	12	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SCPTscheduleDates.start			structure		
SCPTscheduleDates.start.year			signed long	3000	1* 10^0 * (Raw+0)
			-1	-1	1
SCPTscheduleDates.start.month			unsigned short	12	1* 10^0 * (Raw+0)
			(none)	0	1
SCPTscheduleDates.start.day			days_of_month_t	117	none (enumeration)
			-1	-1	1
SCPTscheduleDates.end			structure		
SCPTscheduleDates.end.year			signed long	3000	1* 10^0 * (Raw+0)
			-1	-1	1
SCPTscheduleDates.end.temporary			bitfield	1	1* 10^0 * (Raw+0)
				0	1 (bits: 1) (offset: 0)
SCPTscheduleDates.end.month			bitfield	12	1* 10^0 * (Raw+0)
				0	1 (bits: 7) (offset: 1)

SCPTscheduleDates.end.day	days_of_month_t	117	none (enumeration)
	-1	-1	1
SCPTscheduleDates.qualifier	structure		
SCPTscheduleDates.qualifier.months	months_t	24	none (enumeration)
	-1	-1	1
SCPTscheduleDates.qualifier.days	days_of_month_t	117	none (enumeration)
	-1	-1	1
SCPTscheduleDates.schedule_index	unsigned long	65535	$1* 10^0 * (Raw+0)$
	65535	0	1

5.275 SCPTschedule

Overview: Schedule. Describes the attributes of a daily schedule definition

Index	Size in Bytes	CNP Data Type			
274	3	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SCPTschedule.temporary			bitfield	1	$1* 10^0 * (Raw+0)$
				0	1 (bits: 1) (offset: 0)
SCPTschedule.schedule_priority			bitfield	127	$1* 10^0 * (Raw+0)$
				0	1 (bits: 7) (offset: 1)
SCPTschedule.time_value_index			unsigned long	65535	$1* 10^0 * (Raw+0)$
			65535	0	1

5.276 SCPTscheduleTimeValue

Overview: Schedule time-value pair. Specifies the time and value for a scheduled event

Index	Size in Bytes	CNP Data Type			
275	3	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SCPTscheduleTimeValue.invalid			bitfield	1	1* 10^0 * (Raw+0)
				0	1 (bits: 1) (offset: 0)
SCPTscheduleTimeValue.terminator			bitfield	1	1* 10^0 * (Raw+0)
				0	1 (bits: 1) (offset: 1)
SCPTscheduleTimeValue.hour			bitfield	47	1* 10^0 * (Raw+0)
				0	1 (bits: 6) (offset: 2)
SCPTscheduleTimeValue.minute			unsigned short	59	1* 10^0 * (Raw+0)
			(none)	0	1
SCPTscheduleTimeValue.value			SNVT_sched_val		

5.277 SCPTvalueDefinition

Overview: Value definition. Used to create an array of output values to be used for a schedule. A schedule time-value event specifies a value as an index into a SCPTvalueDefinition array.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
276		inheriting				

5.278 SCPTvalueName

Overview: Value name. Used to create an array of value names for each of the values defined in a SCPTvalueDefinition array

Index	Size in Bytes	CNP Data Type			
277	22	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SCPTvalueName.name			unsigned char	126	$1* 10^0 * (Raw+0)$
			(none)	32	1 array of 22 characters

5.279 SCPTweeklySchedule

Overview: Weekly schedule. Identifies a schedule to be active for each day of the week

Index	Size in Bytes	CNP Data Type			
278	14	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SCPTweeklySchedule.schedule_index			unsigned long	65535	$1* 10^0 * (Raw+0)$
			65535	0	1

5.280 SCPTscheduleName

Overview: Schedule name. Used to create an array of names for schedules defined by a SCPTschedule array or a SCPTscheduleDates array.

Index	Size in Bytes	CNP Data Type			
279	22	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SCPTscheduleName.name			unsigned char	126	$1* 10^0 * (Raw+0)$
			(none)	0	1

5.281 SCPTvalveStroke

Overview: Valve Stroke. This configuration property can be used to provide the stroke to fully open the valve

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
280	2	SNVT_length_mil				

5.282 SCPTvalveNominalSize

Overview: Nominal Valve Size. This configuration property can be used to provide the nominal size of the valve body

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
281	2	SNVT_length_mil				

5.283 SCPTvalveKvs

Overview: Valve Flow. This configuration property can be used to provide the flow through an open valve at 1 bar differential pressure

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
282	2	SNVT_flow_p				

5.284 SCPTvalveType

Overview: Valve Type. This configuration property can be used to provide the valve type

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
283	1	SNVT_dev_c_mode				

5.285 SCPTactuatorCharacteristic

Overview: Actuator Characteristic. This configuration property can be used to provide the characteristic of the actuator

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
284	1	SNVT_dev_c_mode				

5.286 SCPTtrnsTblX2

Overview: Valve-Plug Characteristic Table X. This configuration property will be used in conjunction with the translation-table Y configuration property to create a translation table that dictates how to scale the flow with respect to the mechanical stroke

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
285	30	SNVT_trans_table				

5.287 SCPTtrnsTblY2

Overview: Valve-Plug Characteristic Table Y. This configuration property will be used in conjunction with the translation-table X configuration property to create a translation table that dictates how to scale the flow with respect to the mechanical stroke

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
286	30	SNVT_trans_table				

5.288 SCPTcombFlowCharacteristic

Overview: Combination-Flow Characteristic. This configuration property can be used to provide the desired system control flow characteristic

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
287	1	SNVT_dev_c_mode				

5.289 SCPTtrnsTblX3

Overview: Combination-Flow Characteristic Table X. This configuration property will be used in conjunction with the translation-table Y configuration property to create a translation table that dictates how to scale the flow with respect to the valve capacity

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
288	30	SNVT_trans_table				

5.290 SCPTtrnsTblY3

Overview: Combination-Flow Characteristic Table Y. This configuration property will be used in conjunction with the translation-table X configuration property to create a translation table that dictates how to scale the flow with respect to the valve capacity

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
289	30	SNVT_trans_table				

5.291 SCPTrunTimeAlarm

Overview: Runtime Alarm. This configuration property can be used to provide the alarm threshold for the run time counter output network variable

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
290	7	SNVT_elapsed_tm				

5.292 SCPTtimePeriod

Overview: Historical Period. This input configuration network variable defines the period of time between transfer of a values to the historical register

Index	Size in Bytes	CNP Data Type			
291	2	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SCPTtimePeriod.units			interval_of_month_t	4	none (enumeration)
			-1	-1	1
SCPTtimePeriod.value			union		
SCPTtimePeriod.value.minutes_interval			unsigned short	255	1* 10^0 * (Raw+0)
			(none)	1	1
SCPTtimePeriod.value.date_of_month			unsigned short	31	1* 10^0 * (Raw+0)
			(none)	1	1
SCPTtimePeriod.value.hour_of_day			unsigned short	23	1* 10^0 * (Raw+0)
			(none)	0	1
SCPTtimePeriod.value.day_of_week			days_of_week_t	6	none (enumeration)
			-1	-1	1
SCPTtimePeriod.value.hours_interval			unsigned short	255	1* 10^0 * (Raw+0)
			(none)	1	1

5.293 SCPTpulseValue

Overview: Pulse and Transformer Constant. This configuration property is used to scale the raw pulse value to an energy-meter value

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
292	4	SNVT_muldiv				

5.294 SCPTnumDigits

Overview: Number of Digits on the Meter. This configuration property is used for setting the total number of digits on the meter

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
293	2	SNVT_count				

5.295 SCPTnvPriority

Overview: UNVT Flag. Defines whether the NV is a SNVT (=0) or UNVT (=1)

Index	Size in Bytes	CNP Data Type		
296	1	structure		
Field		CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SCPTnvPriority.user_flag		bitfield	1	$1* 10^0 * (Raw+0)$
			0	1 (bits: 1) (offset: 0)
SCPTnvPriority.nv		bitfield	127	$1* 10^0 * (Raw+0)$
			0	1 (bits: 7) (offset: 1)

5.296 SCPTdefaultSetting

Overview: Default setting.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
297	4	SNVT_setting				

5.297 SCPTlowLimit1Enable

Overview: Low limit 1 Enable. Controls whether low limit 1 is in effect

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
298	1	boolean_t	-1	1	none (enumeration)	-1

5.298 SCPTlowLimit2Enable

Overview: Low limit 2 Enable. Controls whether low limit 2 is in effect

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
299	1	boolean_t	-1	1	none (enumeration)	-1

5.299 SCPTclockCalibration

Overview: Clock Calibration in milliseconds per hour. Corrects clock-cycle variations

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
300	2	signed long	-32768	32767	$1* 10^0 * (Raw+0)$	1 milliseconds per hour

5.300 SCPTneuronId

Overview: Neuron Identifier. A unique 6-byte identifier for a EN 14908-1 device

Index	Size in Bytes	CNP Data Type			
301	6	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SCPTneuronId.id			unsigned char	255	$1* 10^0 * (Raw+0)$
			(none)	0	1

5.301 SCPThighLimit1Enable

Overview: High limit 1 Enable. Controls whether high limit 1 is in effect

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
302	1	boolean_t	-1	1	none (enumeration)	-1

5.302 SCPThighLimit2Enable

Overview: High limit 2 Enable. Controls whether high limit 2 is in effect

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
303	1	boolean_t	-1	1	none (enumeration)	-1

5.303 SCPTahamApplianceModel

Overview: AHAM Appliance Model. Appliance Model code as defined by the Association of Home Appliance Manufacturers

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
304	1	aham_appl_t	-1	6	none (enumeration)	-1

5.304 SCPTdefInput

Overview: .

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
305		inheriting				

5.305 SCPTname1

Overview: Name part 1. Part 1 of the name of the functional block to be used by optional user interface applications. May optionally used with SCPTname2 and SCPTname3. Shall be implemented as a configuration network variable.

Index	Size in Bytes	CNP Data Type			
306	13	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SCPTname1.encoding			char_encoding_t	2	none (enumeration)
			-1	-1	1
SCPTname1.name			unsigned char	255	$1* 10^{0} *$ (Raw+0)
			(none)	0	1

5.306 SCPTscene

Overview: Scene configuration. Scene definition used to create a scene table. This SCPT defines the minimum entries required by the ISI profiles. May be used in combination with SCPTsceneTiming.

Index	Size in Bytes	CNP Data Type			
307	4	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SCPTscene.scene_number			unsigned short	255	$1* 10^{0} *$ (Raw+0)
			0	1	1
SCPTscene.setting			unsigned short	200	$5* 10^{-1} *$ (Raw+0)
			255	0	0,5 % of full level

SCPTscene.rotation	signed short	90	2* 10^0 * (Raw+0)
	-128	-90	2 degrees
SCPTscene.unoccupied_scene	unsigned short	255	1* 10^0 * (Raw+0)
	0	1	1

5.307 SCPTsceneTiming

Overview: Scene timing configuration. Scene timing definition used to supplement a scene table created with a SCPTscene array. This SCPT defines the optional scene table entries for the ISI profiles. When used, it shall be used in combination with a SCPTscene array.

Index	**Size in Bytes**	**CNP Data Type**			
308	4	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SCPTsceneTiming.fade_time			unsigned long	65535	1* 10^{-1} * (Raw+0)
			(none)	0	0,1 s
SCPTsceneTiming.delay_time			unsigned long	65535	1* 10^{-1} * (Raw+0)
			(none)	0	0,1 s

5.308 SCPTname2

Overview: Name part 2. Part 2 of the name of the functional block to be used by optional user interface applications. Shall be used with SCPTname1 and may optionally be used with SCPTname3. This part is concatenated after part 1, and may optionally be followed by part 3. Shall be implemented as a configuration network variable.

Index	**Size in Bytes**	**CNP Data Type**			
309	12	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SCPTname2.name			unsigned char	255	1* 10^0 * (Raw+0)
			(none)	0	1

5.309 SCPTname3

Overview: Name part 3. Part 3 of the name of the functional block to be used by optional user interface applications. Shall be used with SCPTname1 and SCPTname2. This part, if present, is concatenated with parts 1 and 2. Shall be implemented as a configuration network variable.

Index	Size in Bytes	CNP Data Type			
310	12	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SCPTname3.name			unsigned char	255	1* 10^0 * (Raw+0)
			(none)	0	1

5.310 SCPTbuttonPressAction

Overview: Button pressed action. Button action definition used to create a button pressed action array, with an entry per button. This SCPT defines the minimum entries required by the ISI profiles.

Index	Size in Bytes	CNP Data Type			
311	2	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SCPTbuttonPressAction.action			button_action_t	41	none (enumeration)
			-1	-1	1
SCPTbuttonPressAction.setting			union		
			Value for button actions that require a numeric value.		
SCPTbuttonPressAction.setting.scene_number			unsigned short	255	1* 10^0 * (Raw+0)
			0	1	1
SCPTbuttonPressAction.setting.group_number			unsigned short	64	1* 10^0 * (Raw+0)
			255	0	1
SCPTbuttonPressAction.setting.value			unsigned short	200	5* 10^{-1} * (Raw+0)
			255	0	0,5

SCPTbuttonPressAction.setting.angle	signed short	90	1* 10^0 * (Raw+0)
	127	-90	1 degrees

5.311 SCPTbuttonColor

Overview: Button color. Button color configuration for on and off states of a button. May be used to create an array that is used with a SCPTbuttonAction array to specify keypad button behavior.

Index	Size in Bytes	CNP Data Type			
312	6	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SCPTbuttonColor.on_color			structure		
			RGB color for the on state.		
SCPTbuttonColor.on_color.red			unsigned short	255	1* 10^0 * (Raw+0)
			(none)	0	1
SCPTbuttonColor.on_color.green			unsigned short	255	1* 10^0 * (Raw+0)
			(none)	0	1
SCPTbuttonColor.on_color.blue			unsigned short	255	1* 10^0 * (Raw+0)
			(none)	0	1
SCPTbuttonColor.off_color			structure		
			RGB color for the off state		
SCPTbuttonColor.off_color.red			unsigned short	255	1* 10^0 * (Raw+0)
			(none)	0	1
SCPTbuttonColor.off_color.green			unsigned short	255	1* 10^0 * (Raw+0)
			(none)	0	1
SCPTbuttonColor.off_color.blue			unsigned short	255	1* 10^0 * (Raw+0)
			(none)	0	1

5.312 SCPTbuttonRepeatInterval

Overview: Button repeat interval in milliseconds. Time between updates when a button is held down. The updates themselves may be throttled by the application or a SCPTminSendTime CP. Used to create an array used with a SCPTbuttonAction CP array.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
313	2	unsigned long	0	65534	$1* 10^0 * (Raw+0)$	1 milliseconds

5.313 SCPTbuttonHoldAction

Overview: Button held action. Button action definition used to create a button held action array, with an entry per button.

Index	Size in Bytes	CNP Data Type			
314	2	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SCPTbuttonHoldAction.action			button_action_t	41	none (enumeration)
			-1	-1	1
SCPTbuttonHoldAction.setting			union		
			Value for button actions that require a numeric value.		
SCPTbuttonHoldAction.setting.scene_number			unsigned short	255	$1* 10^0 * (Raw+0)$
			0	1	1
SCPTbuttonHoldAction.setting.group_number			unsigned short	64	$1* 10^0 * (Raw+0)$
			255	0	1
SCPTbuttonHoldAction.setting.value			unsigned short	200	$5* 10^{-1} * (Raw+0)$
			255	0	0,5
SCPTbuttonHoldAction.setting.angle			unsigned short	180	$2* 10^0 * (Raw+0)$
			255	0	2 degrees

5.314 SCPTpwrSendOnDelta

Overview: Power send on delta. The minimum change required to propoagate the output value.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
315	2	SNVT_power				

5.315 SCPTsceneName

Overview: Scene name. Name for a scene to be used by optional user interface applications. Used to create an array that supplements a scene table created with a SCPTscene array.

Index	Size in Bytes	CNP Data Type			
316	13	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SCPTsceneName.encoding			char_encoding_t	2	none (enumeration)
			-1	-1	1
SCPTsceneName.name			unsigned short	255	$1* 10^0 * (Raw+0)$
			(none)	0	1

5.316 SCPTmaxPower

Overview: Maximum power. Power level at which the sunblind actuator detects a blocked motor and switches off automatically.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
317	2	SNVT_power				

5.317 SCPTifaceDesc

Overview: Interface description. Human readable description of the interface the functional block is assigned to

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
318	31	SNVT_str_asc				

5.318 SCPTmonInterval

Overview: Monitor Interval. This configuration property defines the interval over which statistics are collected and averages are calculated.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
319	7	SNVT_elapsed_tm				

5.319 SCPTlinkPowerDetectEnable

Overview: Link Power Detection Enabled. Determines, whether link power detection is enabled. If yes, nvoLinkPower indicates existence of link power voltage.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
320	2	SNVT_switch				

5.320 SCPTscanTime

Overview: Scan Time. Duration in which all devices are being queried.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
321	7	SNVT_elapsed_tm				

5.321 SCPTdevListDesc

Overview: Device list entry description. Human readable description for an entry in the device list.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
322	31	SNVT_str_asc				

5.322 SCPTdevListEntry

Overview: Device list entry. Device list entry containing the address of the device to be monitored.

Index	Size in Bytes	CNP Data Type			
323	7	union			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SCPTdevListEntry.address_type			address_type_t	2	none (enumeration)
			-1	-1	1

SCPTdevListEntry.sn	structure		
	This structure is filled out in case the device address is given as subnet/node address		
SCPTdevListEntry.sn.subnet	unsigned short	255	1* 10^0 * (Raw+0)
	(none)	1	1
SCPTdevListEntry.sn.unused	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 0)
SCPTdevListEntry.sn.node	bitfield	127	1* 10^0 * (Raw+0)
		1	1 (bits: 7) (offset: 1)
SCPTdevListEntry.ni	structure		
	This structure is filled out in case the device address is given as unique node ID address		
SCPTdevListEntry.ni.subnet	unsigned short	255	1* 10^0 * (Raw+0)
	(none)	0	1
SCPTdevListEntry.ni.nid	unsigned char	255	1* 10^0 * (Raw+0)
	(none)	0	1

5.323 SCPTlogCapacity

Overview: Data log capacity in bytes. Specifies the total capacity of all data logs on a device. The size of each data log is specified by its cpLogSize value. The value is specified in bytes.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
324	4	unsigned quad	1	4294967295	1* 10^0 * (Raw+0)	1 bytes

5.324 SCPTlogNotificationThreshold

Overview: Data log notification threshold. in percent. Specifies the percentage change in log level required to trigger an update to the Data Log Status (nvoStatus) output.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
325	1	unsigned short	0	200	5* 10^{-1} * (Raw+0)	0,5 percent

5.325 SCPTlogSize

Overview: Log size. in bytes. Capacity of a data log.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
326	4	unsigned quad	0	4294967294	1* 10^0 * (Raw+0)	1 bytes

5.326 SCPTlogType

Overview: Data log type. Specifies the method used to store data in a data log.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
327	1	log_type_t	-1	2	none (enumeration)	-1

5.327 SCPTfanInEnable

Overview: Fan-in enable. Enables fan-in of multiple data sources. When True, the application examines the source address of each input value and uses it to determine the data source of the update.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
328	1	unsigned short	0	1	1* 10^0 * (Raw+0)	1

5.328 SCPTlogTimestampEnable

Overview: Data log enable timestamp. Enables time stamping of each data value. When True, the data logger includes a timestamp of the receipt time for each value received by the data logger.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
329	1	unsigned short	0	1	1* 10^0 * (Raw+0)	1

5.329 SCPTlogHighLimit

Overview: Data log high limit. Enables logging of data greater than or equal to the specified value. All other data are ignored, with the exception that data that are less than or equal to a valid SCPTlogLowLimit value are also logged. All data are logged if the SCPTlogEnableHighLimit and SCPTlogEnableLowLimit values are both invalid.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
330		inheriting				

5.330 SCPTlogLowLimit

Overview: Data log low limit. Enables logging of data less than or equal to the specified value. All other data are ignored, with the exception that data that are greater than or equal to a valid SCPTlogHighLimit value are also logged. All data are logged if the SCPTlogEnableHighLimit and SCPTlogEnableLowLimit values are both invalid.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
331		inheriting				

5.331 SCPTmaxFanIn

Overview: Maximum fan-in. Specifies the maximum number of data sources that may be connected to a network variable. The functional block determines data sources by examining the source address of each update.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
332	2	unsigned long	0	65534	$1* 10^0 * (Raw+0)$	1 data sources

5.332 SCPTlogMinDeltaTime

Overview: Data log minimum delta time in seconds. Minimum amount of time between logged values. This is used to throttle data entry into a data log. When a data value is logged, a subsequent update to the data value is not logged until the time specified by this value has elapsed. If additional updates are received during this time, the older values are discarded and are not stored in the data log. Time of receipt is ignored if the value of this configuration property is zero or invalid.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
333	4	unsigned quad	0	4294967294	$1* 10^0 * (Raw+0)$	1 s

5.333 SCPTlogMinDeltaValue

Overview: Minimum value change between logged values. Used to throttle data entry into a data log. When a data value is logged, a subsequent update to the data value is not logged until the value has changed by a minimum of the amount specified by the delta value.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
334		inheriting				

5.334 SCPTpollRate

Overview: Poll rate. in seconds. Specifies the poll rate for each data source. When this value is greater than zero, the functional block polls each of the data sources identified in the source address array at the rate specified by this value.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
335	4	s32_type (signed 32-bit type)	0	2147483646	$1* 10^{-1} * (Raw+0)$	0,1 s

5.335 SCPTsourceAddress

Overview: Source address. Specifies a source address or element of an array of source addresses for and input to a functional block.

Index	Size in Bytes	CNP Data Type			
336	2	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SCPTsourceAddress.subnet			unsigned short	255	$1* 10^{0} * (Raw+0)$
			0	1	1
SCPTsourceAddress.reserved			bitfield	0	$1* 10^{0} * (Raw+0)$
				0	1 (bits: 1) (offset: 0)
SCPTsourceAddress.node			bitfield	127	$1* 10^{0} * (Raw+0)$
				0	1 (bits: 7) (offset: 1)

5.336 SCPTlogRecord

Overview: Log record. Documents the format of a data log record. Not used as a CP. Data logs have variable sized records--unused fields are not present.

Index	Size in Bytes	CNP Data Type			
337	16	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SCPTlogRecord.timestamp_type			bitfield	3	$1* 10^{0} * (Raw+0)$
				0	1 (bits: 2) (offset: 0)

SCPTlogRecord.record_type	bitfield	3	$1* 10^{0} * (Raw+0)$
		0	1 (bits: 3) (offset: 2)
SCPTlogRecord.multiple_input	bitfield	1	$1* 10^{0} * (Raw+0)$
		0	1 (bits: 1) (offset: 5)
SCPTlogRecord.fan_in	bitfield	1	$1* 10^{0} * (Raw+0)$
		0	1 (bits: 1) (offset: 6)
SCPTlogRecord.point_status	bitfield	1	$1* 10^{0} * (Raw+0)$
		0	1 (bits: 1) (offset: 7)
SCPTlogRecord.data_length	unsigned short	255	$1* 10^{0} * (Raw+0)$
	(none)	0	1 bytes
SCPTlogRecord.time	union		
SCPTlogRecord.time.timestamp	SNVT_time_stamp_p		
SCPTlogRecord.time.offset_stamp	unsigned long	65534	$1* 10^{-2} * (Raw+0)$
	65535	0	0,01 s
SCPTlogRecord.member_index	unsigned short	255	$1* 10^{0} * (Raw+0)$
	0	1	1
SCPTlogRecord.data_source_index	signed long	32767	$1* 10^{0} * (Raw+0)$
	-1	0	1
SCPTlogRecord.data	union		
	Data field for a data log record. Contents depend on record_type value. Unused bytes are not included.		
SCPTlogRecord.data.log_status	log_status_t	5	none (enumeration)
	-1	-1	1
SCPTlogRecord.data.value	s32_type (signed 32-bit type)	2147483647	$1* 10^{0} * (Raw+0)$
	0	-2147483648	1

SCPTlogRecord.data.old_time	SNVT_time_stamp_p		
SCPTlogRecord.point_status_value	point_status_t	4	none (enumeration)
	-1	-1	1

5.337 SCPTlogFileHeader

Overview: Data log header. Describes contents of a data log.

Index	Size in Bytes	CNP Data Type			
338	20	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SCPTlogFileHeader.file_type			unsigned long	2049	1* 10^0 * (Raw+0)
			(none)	2049	1
SCPTlogFileHeader.major_version_number			unsigned short	1	1* 10^0 * (Raw+0)
			(none)	1	1
SCPTlogFileHeader.minor_version_number			unsigned short	0	1* 10^0 * (Raw+0)
			(none)	0	1
SCPTlogFileHeader.log_number			unsigned long	65534	1* 10^0 * (Raw+0)
			65535	0	1
SCPTlogFileHeader.record_count			s32_type (signed 32-bit type)	2147483646	1* 10^0 * (Raw+0)
			2147483647	0	1 s
SCPTlogFileHeader.start_time			SNVT_time_stamp_p		
SCPTlogFileHeader.end_time			SNVT_time_stamp_p		

5.338 SCPTlogAlarmThreshold

Overview: Data log alarm threshold, in percent. Specifies the log level required to trigger an alarm condition for the data logger.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
339	1	unsigned short	0	200	$5* 10^{-1} * (Raw+0)$	0,5 percent

5.339 SCPTlogRequest

Overview: Data log access request. Data log access request message format. Not used as a CP. This request has a variable size--the timestamp field is only included with the get next record request.

Index	Size in Bytes	CNP Data Type			
340	8	structure			
Field		CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution	
SCPTlogRequest.dlap_version		unsigned short	255	$1* 10^{0} * (Raw+0)$	
		0	1	1	
SCPTlogRequest.requested_log		unsigned long	65535	$1* 10^{0} * (Raw+0)$	
		0	1	1	
SCPTlogRequest.last_time		SNVT_time_stamp_p			

5.340 SCPTlogResponse

Overview: Data log access response. Data log access response message format. Not used as a CP. This response has a variable size--unused fields are not included.

Index	Size in Bytes	CNP Data Type			
341	16	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SCPTlogResponse.response			union		
SCPTlogResponse.response.log_record			unsigned short	255	$1* 10^0 * (Raw+0)$
			(none)	0	1
SCPTlogResponse.response.closest_version			unsigned short	255	$1* 10^0 * (Raw+0)$
			(none)	0	1
SCPTlogResponse.response.number_of_logs			unsigned long	65535	$1* 10^0 * (Raw+0)$
			0	1	1

5.341 SCPTlightingGroupEnable

Overview: Lighting group enable. Bit masks to enable or disable up to 64 ISI lighting groups. Group 0 is not used. Groups may also be enabled or disabled using a SNVT_switch_2 update.

Index	Size in Bytes	CNP Data Type			
342	8	structure			
Field			CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SCPTlightingGroupEnable.flags			unsigned short	255	$1* 10^0 * (Raw+0)$
			(none)	0	1

5.342 SCPTsceneColor

Overview: Scene color configuration. Scene color definition used to supplement a scene table created with a SCPTscene array. This SCPT defines optional color scene table entries for the ISI profiles. When used, it shall be used in combination with a SCPTscene array.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
343	5	SNVT_color_2				

5.343 SCPTbkupSchedule

Overview: Backup Schedule.

Index	Size in Bytes	CNP Data Type			
344	4	structure			
Field		**CNP Data Type / Invalid Value**	**Maximum / Minimum**	**Scaled Value / Resolution**	
SCPTbkupSchedule.hour_on		unsigned short	23	$1* 10^0 *$ (Raw+0)	
		255	0	1 Hours	
SCPTbkupSchedule.minute_on		unsigned short	59	$1* 10^0 *$ (Raw+0)	
		(none)	0	1 min	
SCPTbkupSchedule.hour_off		unsigned short	23	$1* 10^0 *$ (Raw+0)	
		255	0	1 Hours	
SCPTbkupSchedule.minute_off		unsigned short	59	$1* 10^0 *$ (Raw+0)	
		(none)	0	1 min	

5.344 SCPTOLCLimits

Overview: OLC Limits Setpoints. MIN/MAX values for the status report are set here.

Index	Size in Bytes	CNP Data Type			
345	22	structure			
Field		**CNP Data Type / Invalid Value**	**Maximum / Minimum**	**Scaled Value / Resolution**	
SCPTOLCLimits.lamp_current_high		SNVT_amp_ac_mil			

SCPTOLCLimits.lamp_current_low	SNVT_amp_ac_mil		
SCPTOLCLimits.main_current_high	SNVT_amp_ac_mil		
SCPTOLCLimits.main_current_low	SNVT_amp_ac_mil		
SCPTOLCLimits.lamp_voltage_high	SNVT_volt_ac		
SCPTOLCLimits.lamp_voltage_low	SNVT_volt_ac		
SCPTOLCLimits.main_voltage_high	SNVT_volt_ac		
SCPTOLCLimits.main_voltage_low	SNVT_volt_ac		
SCPTOLCLimits.power_factor_low	SNVT_pwr_fact		
SCPTOLCLimits.power_high	SNVT_power		
SCPTOLCLimits.power_low	SNVT_power		

5.345 SCPTlampPower

Overview: Lamp Power.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
346	2	SNVT_power				

5.346 SCPTdeviceOutSelection

Overview: Device Output Selection. This selects the Output which will be used on an OLC.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
347	1	olc_select_t	-1	4	none (enumeration)	-1

5.347 SCPTenableStatusMsg

Overview: Enable Status Message.

Index	Size in Bytes	CNP Data Type			
348	5	structure			
Field			**CNP Data Type / Invalid Value**	**Maximum / Minimum**	**Scaled Value / Resolution**
SCPTenableStatusMsg.lamp_current_high			bitfield	1	$1* 10^0 * (Raw+0)$
				0	1 (bits: 1) (offset: 0)
SCPTenableStatusMsg.lamp_current_low			bitfield	1	$1* 10^0 * (Raw+0)$
				0	1 (bits: 1) (offset: 1)
SCPTenableStatusMsg.main_current_high			bitfield	1	$1* 10^0 * (Raw+0)$
				0	1 (bits: 1) (offset: 2)
SCPTenableStatusMsg.main_current_low			bitfield	1	$1* 10^0 * (Raw+0)$
				0	1 (bits: 1) (offset: 3)
SCPTenableStatusMsg.lamp_voltage_high			bitfield	1	$1* 10^0 * (Raw+0)$
				0	1 (bits: 1) (offset: 4)
SCPTenableStatusMsg.lamp_voltage_low			bitfield	1	$1* 10^0 * (Raw+0)$
				0	1 (bits: 1) (offset: 5)
SCPTenableStatusMsg.main_voltage_high			bitfield	1	$1* 10^0 * (Raw+0)$
				0	1 (bits: 1) (offset: 6)
SCPTenableStatusMsg.main_voltage_low			bitfield	1	$1* 10^0 * (Raw+0)$
				0	1 (bits: 1) (offset: 7)

SCPTenableStatusMsg.power_factor_low	bitfield	1	$1* 10^0 *$ (Raw+0)
		0	1 (bits: 1) (offset: 0)
SCPTenableStatusMsg.OLC_temp_high	bitfield	1	$1* 10^0 *$ (Raw+0)
		0	1 (bits: 1) (offset: 1)
SCPTenableStatusMsg.power_high	bitfield	1	$1* 10^0 *$ (Raw+0)
		0	1 (bits: 1) (offset: 2)
SCPTenableStatusMsg.power_low	bitfield	1	$1* 10^0 *$ (Raw+0)
		0	1 (bits: 1) (offset: 3)
SCPTenableStatusMsg.relay_failure	bitfield	1	$1* 10^0 *$ (Raw+0)
		0	1 (bits: 1) (offset: 4)
SCPTenableStatusMsg.cap_failure	bitfield	1	$1* 10^0 *$ (Raw+0)
		0	1 (bits: 1) (offset: 5)
SCPTenableStatusMsg.lamp_failure	bitfield	1	$1* 10^0 *$ (Raw+0)
		0	1 (bits: 1) (offset: 6)
SCPTenableStatusMsg.ballast_failure	bitfield	1	$1* 10^0 *$ (Raw+0)
		0	1 (bits: 1) (offset: 7)
SCPTenableStatusMsg.inter_com_failure	bitfield	1	$1* 10^0 *$ (Raw+0)
		0	1 (bits: 1) (offset: 0)
SCPTenableStatusMsg.exter_com_failure	bitfield	1	$1* 10^0 *$ (Raw+0)
		0	1 (bits: 1) (offset: 1)
SCPTenableStatusMsg.main_volt_below_spec	bitfield	1	$1* 10^0 *$ (Raw+0)
		0	1 (bits: 1) (offset: 2)
SCPTenableStatusMsg.lamp_restart_count	bitfield	1	$1* 10^0 *$ (Raw+0)
		0	1 (bits: 1) (offset: 3)
SCPTenableStatusMsg.fading_ready	bitfield	1	$1* 10^0 *$ (Raw+0)
		0	1 (bits: 1) (offset: 4)

SCPTenableStatusMsg.ballast_temp_high	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 5)
SCPTenableStatusMsg.digi_in_A	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 6)
SCPTenableStatusMsg.digi_in_B	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 7)
SCPTenableStatusMsg.bit_25_res	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 0)
SCPTenableStatusMsg.bit_26_res	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 1)
SCPTenableStatusMsg.bit_27_res	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 2)
SCPTenableStatusMsg.bit_28_res	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 3)
SCPTenableStatusMsg.bit_29_res	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 4)
SCPTenableStatusMsg.bit_30_res	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 5)
SCPTenableStatusMsg.bit_31_res	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 6)
SCPTenableStatusMsg.bit_32_res	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 7)
SCPTenableStatusMsg.bit_33_res	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 0)
SCPTenableStatusMsg.bit_34_res	bitfield	1	1* 10^0 * (Raw+0)
		0	1 (bits: 1) (offset: 1)

SCPTenableStatusMsg.bit_35_res	bitfield	1	$1* 10^0$ * (Raw+0)
		0	1 (bits: 1) (offset: 2)
SCPTenableStatusMsg.bit_36_res	bitfield	1	$1* 10^0$ * (Raw+0)
		0	1 (bits: 1) (offset: 3)
SCPTenableStatusMsg.bit_37_res	bitfield	1	$1* 10^0$ * (Raw+0)
		0	1 (bits: 1) (offset: 4)
SCPTenableStatusMsg.bit_38_res	bitfield	1	$1* 10^0$ * (Raw+0)
		0	1 (bits: 1) (offset: 5)
SCPTenableStatusMsg.bit_39_res	bitfield	1	$1* 10^0$ * (Raw+0)
		0	1 (bits: 1) (offset: 6)
SCPTenableStatusMsg.bit_40_res	bitfield	1	$1* 10^0$ * (Raw+0)
		0	1 (bits: 1) (offset: 7)

5.348 SCPTmaxLevelVolt

Overview: Maximum Dim Voltage.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
349	2	SNVT_volt				

5.349 SCPTgeoLocation

Overview: Geographic Location.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
350	31	SNVT_geo_loc				

5.350 SCPTprogName

Overview: Program Name. Name of currently loaded program

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
351	31	SNVT_str_asc				

5.351 SCPTprogRevision

Overview: Program Revision. Revision number and date of currently loaded program.

Index	Size in Bytes	CNP Data Type			
352	11	structure			
Field			**CNP Data Type / Invalid Value**	**Maximum / Minimum**	**Scaled Value / Resolution**
SCPTprogRevision.major_version			unsigned short	255	$1* 10^0 * (Raw+0)$
			(none)	0	1
SCPTprogRevision.minor_version			unsigned short	255	$1* 10^0 * (Raw+0)$
			(none)	0	1
SCPTprogRevision.build_number			unsigned long	65535	$1* 10^0 * (Raw+0)$
			(none)	0	1
SCPTprogRevision.build_date			SNVT_time_stamp		

5.352 SCPTprogSelect

Overview: Program Select. Buffer Id where the currently loaded program is stored.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
353	1	unsigned short	0	255	$1* 10^0 * (Raw+0)$	1

5.353 SCPTprogSourceLocation

Overview: Source Location. Location from where the current program was downloaded.

Index	Size in Bytes	CNP Data Type			
354	255	structure			
Field			**CNP Data Type / Invalid Value**	**Maximum / Minimum**	**Scaled Value / Resolution**
SCPTprogSourceLocation.location			unsigned char	255	$1* 10^0 * (Raw+0)$
			(none)	0	1

5.354 SCPTprogFileIndexes

Overview: File Indexes. Indexes of first and last LonMark files where programs may be stored.

Index	Size in Bytes	CNP Data Type	
355	2	structure	

Field	CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SCPTprogFileIndexes.first_file_index	unsigned short	255	$1* 10^0 * (Raw+0)$
	(none)	3	1
SCPTprogFileIndexes.last_file_index	unsigned short	255	$1* 10^0 * (Raw+0)$
	(none)	3	1

5.355 SCPTprogCmdHistory

Overview: Command History. Log of recent commands, with time stamp.

Index	Size in Bytes	CNP Data Type	
356	88	structure	

Field	CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution
SCPTprogCmdHistory.timestamp	SNVT_time_stamp		
SCPTprogCmdHistory.command	object_request_t	24	none (enumeration)
	-1	-1	1
SCPTprogCmdHistory.description	unsigned char	255	$1* 10^0 * (Raw+0)$
	(none)	0	1

5.356 SCPTprogStateHistory

Overview: State History. Log of recent status values, with time stamp.

Index	Size in Bytes	CNP Data Type	
357	8	structure	

Field	CNP Data Type / Invalid Value	Maximum / Minimum	Scaled Value / Resolution

SCPTprogStateHistory.time_of_state_change	SNVT_time_stamp		
SCPTprogStateHistory.state	program_state_t	5	none (enumeration)
	-1	-1	1

5.357 SCPTnsdsFbIndex

Overview: Index of Functional Block.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaling	Resolution
358	2	unsigned long	0	65535	$1* 10^0 * (Raw+0)$	1

5.358 SCPTcurrentSenseEnable

Overview: Current sense enable. Enables current sensing for manual load control.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
359	1	boolean_t	-1	1	none (enumeration)	-1

5.359 SCPTmeasurementInterval

Overview: Measurement interval. Time period used for a measurement.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
360	2	SNVT_time_sec				

5.360 SCPTlightingGroupMembership

Overview: Lighting group membership. Bit masks to specify membership in up to 64 ISI lighting groups. Group 0 is not used.

Index	Size in Bytes	CNP Data Type		
361	8	structure		
Field		**CNP Data Type / Invalid Value**	**Maximum / Minimum**	**Scaled Value / Resolution**
SCPTlightingGroupMembership.flags		unsigned short	255	$1* 10^0 * (Raw+0)$
		(none)	0	1

5.361 SCPTloadControlOffset

Overview: Load control offsets. Offsets to be used during standby (unoccupied state but home, or sleep mode) and demand-response modes.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
362	6	SNVT_load_offsets				

5.362 SCPTprogErrorHistory

Overview: State History. Log of recent status values, with time stamp

Index	Size in Bytes	CNP Data Type			
363	8	structure			
Field		**CNP Data Type / Invalid Value**	**Maximum / Minimum**	**Scaled Value / Resolution**	
SCPTprogErrorHistory.time_of_error		SNVT_time_stamp			
SCPTprogErrorHistory.error		program_status_error_t	63	none (enumeration)	
		-1	-1	1	

5.363 SCPTnvUsage

Overview: NV usage. The SCPTnvUsage CPs shall be used to indicate whether the NVs are in use by the loaded program.

Index	Size in Bytes	CNP Data Type			
364	1	structure			
Field		**CNP Data Type / Invalid Value**	**Maximum / Minimum**	**Scaled Value / Resolution**	
SCPTnvUsage.in_use		bitfield	1	$1* 10^0 * (Raw+0)$	
			0	1 (bits: 1) (offset: 0)	
SCPTnvUsage.mfg		bitfield	1	$1* 10^0 * (Raw+0)$	
			0	1 (bits: 7) (offset: 1)	

5.364 SCPTscheduleSunday

Overview: Sunday schedule. A structure containing an array of seven time-value pairs that specify the daily schedule for Sunday; unused time-value pairs have an invalid value (31) for the hour; if two time-value pairs specify the same time, the first with a valid output value is used.

Index	Size in Bytes	CNP Data Type			
365	28	structure			
Field		**CNP Data Type / Invalid Value**	**Maximum / Minimum**	**Scaled Value / Resolution**	
SCPTscheduleSunday.time_value		SNVT_time_val_2			

5.365 SCPTscheduleMonday

Overview: Monday schedule. A structure containing an array of seven time-value pairs that specify the daily schedule for Monday; unused time-value pairs have an invalid value (31) for the hour; if two time-value pairs specify the same time, the first with a valid output value is used.

Index	Size in Bytes	CNP Data Type			
366	28	structure			
Field		**CNP Data Type / Invalid Value**	**Maximum / Minimum**	**Scaled Value / Resolution**	
SCPTscheduleMonday.time_value		SNVT_time_val_2			

5.366 SCPTscheduleTuesday

Overview: Tuesday schedule. A structure containing an array of seven time-value pairs that specify the daily schedule for Tuesday; unused time-value pairs have an invalid value (31) for the hour; if two time-value pairs specify the same time, the first with a valid output value is used.

Index	Size in Bytes	CNP Data Type			
367	28	structure			
Field		**CNP Data Type / Invalid Value**	**Maximum / Minimum**	**Scaled Value / Resolution**	
SCPTscheduleTuesday.time_value		SNVT_time_val_2			

5.367 SCPTscheduleWednesday

Overview: Wednesday schedule. A structure containing an array of seven time-value pairs that specify the daily schedule for Wednesday; unused time-value pairs have an invalid value (31) for the hour; if two time-value pairs specify the same time, the first with a valid output value is used.

Index	Size in Bytes	CNP Data Type			
368	28	structure			
Field			**CNP Data Type / Invalid Value**	**Maximum / Minimum**	**Scaled Value / Resolution**
SCPTscheduleWednesday.time_value			SNVT_time_val_2		

5.368 SCPTscheduleThursday

Overview: Thursday schedule. A structure containing an array of seven time-value pairs that specify the daily schedule for Thursday; unused time-value pairs have an invalid value (31) for the hour; if two time-value pairs specify the same time, the first with a valid output value is used.

Index	Size in Bytes	CNP Data Type			
369	28	structure			
Field			**CNP Data Type / Invalid Value**	**Maximum / Minimum**	**Scaled Value / Resolution**
SCPTscheduleThursday.time_value			SNVT_time_val_2		

5.369 SCPTscheduleFriday

Overview: Friday schedule. A structure containing an array of seven time-value pairs that specify the daily schedule for Friday; unused time-value pairs have an invalid value (31) for the hour; if two time-value pairs specify the same time, the first with a valid output value is used.

Index	Size in Bytes	CNP Data Type			
370	28	structure			
Field			**CNP Data Type / Invalid Value**	**Maximum / Minimum**	**Scaled Value / Resolution**
SCPTscheduleFriday.time_value			SNVT_time_val_2		

5.370 SCPTscheduleSaturday

Overview: Saturday schedule. A structure containing an array of seven time-value pairs that specify the daily schedule for Saturday; unused time-value pairs have an invalid value (31) for the hour; if two time-value pairs specify the same time, the first with a valid output value is used.

Index	Size in Bytes	CNP Data Type			
371	28	structure			
Field			**CNP Data Type / Invalid Value**	**Maximum / Minimum**	**Scaled Value / Resolution**
SCPTscheduleSaturday.time_value			SNVT_time_val_2		

5.371 SCPToccupancyBehavior

Overview: Occupancy behavior. Specifies mapping of scheduled occupancy values to primary occupancy states based on local occupancy inputs.

Index	Size in Bytes	CNP Data Type			
372	5	structure			
Field			**CNP Data Type / Invalid Value**	**Maximum / Minimum**	**Scaled Value / Resolution**
SCPToccupancyBehavior.ob_nul_value			occup_t	3	none (enumeration)
			-1	-1	1
SCPToccupancyBehavior.ob_occupied_value			occup_t	3	none (enumeration)
			-1	-1	1
SCPToccupancyBehavior.ob_unoccupied_value			occup_t	3	none (enumeration)
			-1	-1	1
SCPToccupancyBehavior.ob_bypass_value			occup_t	3	none (enumeration)
			-1	-1	1
SCPToccupancyBehavior.ob_standby_value			occup_t	3	none (enumeration)
			-1	-1	1

5.372 SCPTtimeSource

Overview: Time source. Specifies the source of time.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
373	1	time_source_t	-1	4	none (enumeration)	-1

5.373 SCPTscheduleException

Overview: Exception schedule. An event that overrides a daily schedule; typically used for special events.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
374	8	SNVT_sched_exc				

5.374 SCPTscheduleHoliday

Overview: Holiday or vacation schedule. An event that overrides a daily schedule; typically used for holiday or vacation event.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
375	8	SNVT_sched_exc				

5.375 SCPTrandomizationInterval

Overview: Randomization interval. Specifies an interval around a scheduled time that is used by a scheduler to calculate a random event time. Used to reduce simultaneous startup and shutdown of many devices by multiple schedulers.

Index	Size in Bytes	CNP Data Type			
376	4	structure			
Field			**CNP Data Type / Invalid Value**	**Maximum / Minimum**	**Scaled Value / Resolution**
SCPTrandomizationInterval.earliest_time_offset			SNVT_time_sec		
SCPTrandomizationInterval.latest_time_offset			SNVT_time_sec		

5.376 SCPTsunriseTime

Overview: Sunrise time. Time used for sunrise-relative scheduling; shall be implemented as a configuration network variable; only the time fields are used for scheduling--the date fields indicate the date used for the configured time.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
377	7	SNVT_time_stamp				

5.377 SCPTsunsetTime

Overview: Sunset time. Time used for sunset-relative scheduling; shall be implemented as a configuration network variable; only the time fields are used for scheduling--the date fields indicate the date used for the configured time.

Index	Size in Bytes	CNP Data Type	Minimum	Maximum	Scaled Value	Invalid
378	7	SNVT_time_stamp				

5.378 SCPTschedulerOptions

Overview: Scheduler options.

Index	Size in Bytes	CNP Data Type			
379	1	structure			
Field			**CNP Data Type / Invalid Value**	**Maximum / Minimum**	**Scaled Value / Resolution**
SCPTschedulerOptions.reserved			bitfield	1	$1* 10^0 * (Raw+0)$
				0	1 (bits: 5) (offset: 0)
SCPTschedulerOptions.alternate_time_source			bitfield	1	$1* 10^0 * (Raw+0)$
				0	1 (bits: 1) (offset: 5)
SCPTschedulerOptions.general_purpose_output			bitfield	1	$1* 10^0 * (Raw+0)$
				0	1 (bits: 1) (offset: 6)
SCPTschedulerOptions.sunrise_sunset_relative			bitfield	1	$1* 10^0 * (Raw+0)$
				0	1 (bits: 1) (offset: 7)

5.379 SCPToccupancyThresholds

Overview: Occupancy thresholds. Specifies the minimum number of occupancy sensors that shall report the same value to override a scheduled output value.

Index	Size in Bytes	CNP Data Type			
380	3	structure			
Field			**CNP Data Type / Invalid Value**	**Maximum / Minimum**	**Scaled Value / Resolution**
SCPToccupancyThresholds.occupied			unsigned short	50	$1* 10^0$ * (Raw+0)
			255	0	1
SCPToccupancyThresholds.standby_to_occupied			unsigned short	50	$1* 10^0$ * (Raw+0)
			255	0	1
SCPToccupancyThresholds.unoccupied_to_occupied			unsigned short	50	$1* 10^0$ * (Raw+0)
			255	0	1

6 Standard Enumeration Type

6.1 Introduction

Standard enumeration types facilitate interoperability by providing standard definitions for enumerated values used for communication between devices or for configuring a device.

This document provides information on all available standard enumeration types. A standard enumeration type index is defined for each standard enumeration type that is used when defining enumeration type references in SNVT and SCPT definitions. The enumeration type names are provided for use in network and development tools.

6.2 days_of_week_t

Index: 1

DAY_NUL:	-1	Invalid Value
DAY_SUN:	0	Sunday
DAY_MON:	1	Monday
DAY_TUE:	2	Tuesday
DAY_WED:	3	Wednesday
DAY_THU:	4	Thursday
DAY_FRI:	5	Friday
DAY_SAT:	6	Saturday

6.3 discrete_levels_t

Index: 2

ST_NUL:	-1
ST_OFF:	0
ST_LOW:	1
ST_MED:	2
ST_HIGH:	3
ST_ON:	4

6.4 telcom_states_t

Index: 3

TEL_NUL:	-1	Invalid Value
TEL_NOTINUSE:	0	"Null State (U0)" not in use
TEL_OFFHOOK:	1	"Call Initiated (U1)"
TEL_DIALING:	2	"Overlap Sending (U2)"
TEL_DIALCOMP:	3	"Outgoing Call Proceeding (U3)"
TEL_RINGBACK:	4	"Call Delivered (U4)" hearing ringback
TEL_INCOMING:	5	"Call Present (U6)" incoming call has not yet started ringing (only on ISDN line)
TEL_RINGING:	6	"Call Received (U7)" incoming call when the user has indicated alerting but has not yet answered
TEL_ANSWERED:	7	"Connect Request (U8)" user has answered the call and is waiting to be awarded the call
TEL_CONNECTED:	8	
TEL_TALKING:	9	"Active (U10)" two parties are exchanging data
TEL_HANGINGUP:	10	"Disconnect Request (U11)" user has hung up
TEL_HUNGUPX:	11	"Disconnect Indication (U12)" the other side hung up
TEL_HOLD:	12	"Suspend Request (U15)" user has requested the network suspend the call
TEL_UNHOLD:	13	"Resume Request (U17)" resume a held call (usually go back to TEL_TALKING)
TEL_RELEASE:	14	"Release Request (U19)" user has requested the network to release
TEL_FULLDUP:	15	"Overlap Receiving (U25)" user has acknowledged the call and is prepared to receive additional
TEL_BLOCKED:	16	connection with blocking (call-waiting disabled)
TEL_CWAIT:	17	call-waiting coming in
TEL_DESTBUSY:	18	destination busy
TEL_NETBUSY:	19	problem, network
TEL_ERROR:	20	problem, non-network

6.5 config_source_t

Index: 4

CFG_NUL:	-1	Invalid Value
CFG_LOCAL:	0	Device will use self-installation functions to set its own network image
CFG_EXTERNAL:	1	Device's network image will be set by an outside source

6.6 file_request_t

Index: 5

FR_NUL:	-1	Invalid Value
FR_OPEN_TO_SEND:	0	Sequential access read
FR_OPEN_TO_RECEIVE:	1	Sequential access write
FR_CLOSE_FILE:	2	Close and save file
FR_CLOSE_DELETE_FILE:	3	Close and delete file
FR_DIRECTORY_LOOKUP:	4	Retrieve directory entry
FR_OPEN_TO_SEND_RA:	5	Random access read
FR_OPEN_TO_RECEIVE_RA:	6	Random access write

6.7 file_status_t

Index: 6

FS_NUL:	-1	Invalid Value
FS_XFER_OK:	0	File transfer successful
FS_LOOKUP_OK:	1	Directory lookup successful
FS_OPEN_FAIL:	2	Error on opening file
FS_LOOKUP_ERR:	3	Error on directory lookup
FS_XFER_UNDERWAY:	4	File transfer in progress
FS_IO_ERR:	5	Error on reading/writing file
FS_TIMEOUT_ERR:	6	File transfer timed out
FS_WINDOW_ERR:	7	Window sequence error
FS_AUTH_ERR:	8	Authentication failure
FS_ACCESS_UNAVAIL:	9	Access mode not supported

FS_SEEK_INVALID:	10	Random access beyond EOF
FS_SEEK_WAIT:	11	

6.8 alarm_type_t

Index: 7

AL_HEADER:	-13	Update sequence header
AL_FOOTER:	-12	Update sequence footer
AL_DEBUG:	-11	Debug information (not an alarm)
AL_INFO:	-10	Information update (not an alarm)
AL_SYSTEM_INFO:	-6	System information (not an alarm)
AL_VALUE_INVALID:	-5	The value is invalid
AL_CONSTANT:	-4	The value is a constant value (not an alarm)
AL_OFFLINE:	-3	The device is offline
AL_UNKNOWN:	-2	Alarm condition unknown (may be due to a communication failure or hardware failure)
AL_NUL:	-1	Invalid alarm type value (alarm condition not specified)
AL_NO_CONDITION:	0	No alarm condition present
AL_ALM_CONDITION:	1	Unspecified alarm condition present
AL_TOT_SVC_ALM_1:	2	Total/service interval alarm 1 (component requires service or maintenance)
AL_TOT_SVC_ALM_2:	3	Total/service interval alarm 2
AL_TOT_SVC_ALM_3:	4	Total/service interval alarm 3
AL_LOW_LMT_CLR_1:	5	Alarm low limit alarm clear 1
AL_LOW_LMT_CLR_2:	6	Alarm low limit alarm clear 2
AL_HIGH_LMT_CLR_1:	7	Alarm high limit alarm clear 1
AL_HIGH_LMT_CLR_2:	8	Alarm high limit alarm clear 2
AL_LOW_LMT_ALM_1:	9	Alarm low limit alarm 1
AL_LOW_LMT_ALM_2:	10	Alarm low limit alarm 2
AL_HIGH_LMT_ALM_1:	11	Alarm high limit alarm 1
AL_HIGH_LMT_ALM_2:	12	Alarm high limit alarm 2
AL_FIR_ALM:	13	Fire alarm condition

AL_FIR_PRE_ALM:	14	Fire pre-alarm condition
AL_FIR_TRBL:	15	Fire-related trouble (fault) condition
AL_FIR_SUPV:	16	Fire-related supervisory condition (e.g., sprinkler pressure)
AL_FIR_TEST_ALM:	17	Fire-related test-mode alarm condition
AL_FIR_TEST_PRE_ALM:	18	Fire-related test-mode pre-alarm condition
AL_FIR_ENVCOMP_MAX:	19	Fire-related maximum environmental compensation level reached
AL_FIR_MONITOR_COND:	20	Fire-related abnormal input condition
AL_FIR_MAINT_ALERT:	21	Fire-related maintenance alert
AL_FATAL_ERROR:	30	Fatal application error
AL_ERROR:	31	Other error condition
AL_WARNING:	32	Other warning condition

6.9 priority_level_t

Index: 8

PR_NUL:	-1	Invalid Value
PR_LEVEL_0:	0	Lowest alarm priority level
PR_LEVEL_1:	1	
PR_LEVEL_2:	2	
PR_LEVEL_3:	3	Highest alarm priority level
PR_1:	4	Life Safety Fire Alarms (BACnet Priority 2)
PR_2:	5	Property Safety Fire Alarms (BACnet Priority 3)
PR_3:	6	Fire Supervisory Alarm (BACnet Priority 4)
PR_4:	7	Fire Trouble/Fault (Display) (BACnet Priority 5)
PR_6:	8	Fire Pre-Alarm, HVAC Critical Equipment Alarm (BACnet Priority 6)
PR_8:	9	HVAC Alarms (BACnet Priority 8)
PR_10:	10	HVAC Critical Equipment RTN, Fire RTN (Display) (BACnet Priority 10)
PR_16:	11	HVAC RTN (lowest priority) (BACnet Priority 16)

6.10 currency_t

Index: 9

CU_NUL:	-1	Invalid Value
CU_ARGENTINA_PESO:	0	Argentine Peso
CU_AUSTRALIA_DOLLAR:	1	Australian Dollar
CU_AUSTRIA_SCHILLING:	2	Austrian Schilling
CU_BAHRAIN_DINAR:	3	Bahraini Dinar
CU_BELGIUM_FRANC:	4	Belgian Franc
CU_BRAZIL_CRUZEIRO_REAL:	5	Brazilian Cruzeiro Real
CU_BRITAIN_POUND:	6	British Pound
CU_CANADA_DOLLAR:	7	Canadian Dollar
CU_CZECH_KORUNA:	8	Czechoslovakian Koruna
CU_CHILE_PESO:	9	Chilean Peso
CU_CHINA_RENMINBI:	10	Chinese Renminbi Yuan
CU_COLOMBIA_PESO:	11	Colombian Peso
CU_DENMARK_KRONE:	12	Danish Krone
CU_ECUADOR_SUCRE:	13	Ecuadorian Sucre
CU_EUROPEAN_CURRENCY_UNIT:	14	European Euro
CU_FINLAND_MARKKA:	15	Finnish Markka
CU_FRANCE_FRANC:	16	French Franc
CU_GERMANY_MARK:	17	German Mark
CU_GREECE_DRACHMA:	18	Greek Drachma
CU_HONG_KONG_DOLLAR:	19	Hong Kong Dollar
CU_HUNGARY_FORINT:	20	Hungarian Forint
CU_INDIA_RUPEE:	21	Indian Rupee
CU_INDONESIA_RUPIAH:	22	Indonesian Rupiah
CU_IRELAND_PUNT:	23	Irish Punt
CU_ISRAEL_SHEKEL:	24	Israeli Shekel
CU_ITALY_LIRA:	25	Italian Lira

CU_JAPAN_YEN:	26	Japanese Yen
CU_JORDAN_DINAR:	27	Jordanian Dinar
CU_KUWAIT_DINAR:	28	Kuwaiti Dinar
CU_LEBANON_POUND:	29	Lebanese Pound
CU_MALAYSIA_RINGGIT:	30	Malaysian Ringgit
CU_MALTA_LIRA:	31	Maltese Lira
CU_MEXICO_PESO:	32	Mexican New Peso
CU_NETHERLANDS_GUILDER:	33	Netherlands Guilder
CU_NEW_ZEALAND_DOLLAR:	34	New Zealand Dollar
CU_NORWAY_KRONE:	35	Norwegian Krone
CU_PAKISTAN_RUPEE:	36	Pakistani Rupee
CU_PERU_NEW_SOL:	37	Peruvian New Sol
CU_PHILIPPINES_PESO:	38	Philippine Peso
CU_POLAND_ZLOTY:	39	Polish Zloty
CU_PORTUGAL_ESCUDO:	40	Portuguese Escudo
CU_SAUDI_ARABIA_RIYAL:	41	Saudi Arabian Riyal
CU_SINGAPORE_DOLLAR:	42	Singaporean Dollar
CU_SLOVAK_KORUNA:	43	Slavic Koruna
CU_SOUTH_AFRICA_RAND:	44	South African Rand
CU_SOUTH_KOREA_WON:	45	South Korean Won
CU_SPAIN_PESETA:	46	Spanish Peseta
CU_SPECIAL_DRAWING_RIGHTS:	47	international governmental exchange
CU_SWEDEN_KRONA:	48	Swedish Krona
CU_SWITZERLAND_FRANC:	49	Swiss Franc
CU_TAIWAN_DOLLAR:	50	Taiwanese Dollar
CU_THAILAND_BAHT:	51	Thai Baht
CU_TURKEY_LIRA:	52	Turkish Lira
CU_UNITED_ARAB_DIRHAM:	53	United Arab Emirates Dirham
CU_UNITED_STATES_DOLLAR:	54	United States Dollar

CU_URUGUAY_NEW_PESO:	55	Uruguayan New Peso
CU_VENEZUELA_BOLIVAR:	56	Venezuelan Bolivar

6.11 object_request_t

Index: 10

RQ_NUL:	-1	Invalid Value
RQ_NORMAL:	0	Enable object and remove override
RQ_DISABLED:	1	Disable object
RQ_UPDATE_STATUS:	2	Report object status
RQ_SELF_TEST:	3	Perform object self-test
RQ_UPDATE_ALARM:	4	Update alarm status
RQ_REPORT_MASK:	5	Report status bit mask
RQ_OVERRIDE:	6	Override object
RQ_ENABLE:	7	Enable object
RQ_RMV_OVERRIDE:	8	Remove object override
RQ_CLEAR_STATUS:	9	Clear object status
RQ_CLEAR_ALARM:	10	Clear object alarm
RQ_ALARM_NOTIFY_ENABLED:	11	Enable alarm notification
RQ_ALARM_NOTIFY_DISABLED:	12	Disable alarm notification
RQ_MANUAL_CTRL:	13	Enable object for manual control
RQ_REMOTE_CTRL:	14	Enable object for remote control
RQ_PROGRAM:	15	Enable programming of special configuration properties
RQ_CLEAR_RESET:	16	Clear reset-complete flag (reset_complete)
RQ_RESET:	17	Execute reset-sequence of object
RQ_CLEAR_LOG:	18	Clear data log
RQ_LOAD_PROGRAM:	19	Load the program specified in SCPTprogSelect
RQ_RUN_PROGRAM:	20	Run the currently loaded program. If the program was halted manually, this will resume running from the point is was halted.
RQ_HALT_PROGRAM:	21	Halt the currently loaded program. This will preserve the program state and a subsequent Run command will resume the program from where it was halted.

RQ_RESTART_PROGRAM:	22	Restart the currently loaded program from the beginning.
RQ_UNLOAD_PROGRAM:	23	Unload the currently loaded program
RQ_STEP_PROGRAM:	24	Executes the next logical operation (line, statement, instruction, logic block, etc.) of the currently loaded program. The program state shall be "idle" or "halted" to accept this command, otherwise it will be ignored. The program returns to "halted" state after execution of this command

6.12 learn_mode_t

Index: 11

LN_NUL:	-1	Invalid Value
LN_RECALL:	0	Recall
LN_LEARN_CURRENT:	1	Learn present value
LN_LEARN_VALUE:	2	Learn given value
LN_REPORT_VALUE:	3	Report the value

6.13 override_t

Index: 12

OV_NUL:	-1	Invalid Value
OV_RETAIN:	0	Retain current level
OV_SPECIFIED:	1	Go to specified level
OV_DEFAULT:	2	Go to default level

6.14 emerg_t

Index: 13

EMERG_NUL:	-1	Invalid Value
EMERG_NORMAL:	0	No emergency mode
EMERG_PRESSURIZE:	1	Emergency pressurize mode
EMERG_DEPRESSURIZE:	2	Emergency depressurize mode
EMERG_PURGE:	3	Emergency purge mode
EMERG_SHUTDOWN:	4	Emergency shutdown mode
EMERG_FIRE:	5	Emergency fire mode

6.15 hvac_t

Index: 14

HVAC_NUL:	-1	Invalid value
HVAC_AUTO:	0	Controller automatically changes between application modes
HVAC_HEAT:	1	Heating only
HVAC_MRNG_WRMUP:	2	Application-specific morning warm-up
HVAC_COOL:	3	Cooling only
HVAC_NIGHT_PURGE:	4	Application-specific night purge
HVAC_PRE_COOL:	5	Application-specific pre-cool
HVAC_OFF:	6	Controller not controlling outputs
HVAC_TEST:	7	Equipment being tested
HVAC_EMERG_HEAT:	8	Emergency heat mode (heat pump)
HVAC_FAN_ONLY:	9	Air not conditioned, fan turned on
HVAC_FREE_COOL:	10	Cooling with compressor not running
HVAC_ICE:	11	Ice-making mode
HVAC_MAX_HEAT:	12	Maximum heating mode
HVAC_ECONOMY:	13	Economic Heat/Cool mode
HVAC_DEHUMID:	14	Dehumidification mode
HVAC_CALIBRATE:	15	Calibration mode
HVAC_EMERG_COOL:	16	Emergency cool mode
HVAC_EMERG_STEAM:	17	Emergency steam mode
HVAC_MAX_COOL:	18	
HVAC_HVC_LOAD:	19	
HVAC_NO_LOAD:	20	

6.16 occup_t

Index: 15

OC_NUL:	-1	Invalid Value
OC_OCCUPIED:	0	Area is occupied
OC_UNOCCUPIED:	1	Area is unoccupied
OC_BYPASS:	2	Area is temporarily occupied for the bypass period
OC_STANDBY:	3	Area is temporarily unoccupied

6.17 hvac_overid_t

Index: 16

HVO_NUL:	-1	Invalid Value
HVO_OFF:	0	Not overridden
HVO_POSITION:	1	
HVO_FLOW_VALUE:	2	Override flow in l/s - use flow field
HVO_FLOW_PERCENT:	3	Override flow percentage - use percent field
HVO_OPEN:	4	Override to position = 100%
HVO_CLOSE:	5	Override to position = 0%
HVO_MINIMUM:	6	Override to configured minimum
HVO_MAXIMUM:	7	Override to configured maximum
HVO_UNUSED8:	8	
HVO_UNUSED9:	9	
HVO_UNUSED10:	10	
HVO_UNUSED11:	11	
HVO_UNUSED12:	12	
HVO_UNUSED13:	13	
HVO_UNUSED14:	14	
HVO_UNUSED15:	15	
HVO_UNUSED16:	16	
HVO_POSITION_1:	17	

HVO_FLOW_VALUE_1:	18	Override flow in liters/sec - use flow field
HVO_FLOW_PERCENT_1:	19	Override flow percentage - use percent field
HVO_OPEN_1:	20	Override to position = 100%
HVO_CLOSE_1:	21	Override to position = 0%
HVO_MINIMUM_1:	22	Override to configured minimum
HVO_MAXIMUM_1:	23	Override to configured maximum
HVO_UNUSED24:	24	
HVO_UNUSED25:	25	
HVO_UNUSED26:	26	
HVO_UNUSED27:	27	
HVO_UNUSED28:	28	
HVO_UNUSED29:	29	
HVO_UNUSED30:	30	
HVO_UNUSED31:	31	
HVO_UNUSED32:	32	
HVO_POSITION_2:	33	
HVO_FLOW_VALUE_2:	34	Override flow in liters/sec - use flow field
HVO_FLOW_PERCENT_2:	35	Override flow percentage - use percent field
HVO_OPEN_2:	36	Override to position = 100%
HVO_CLOSE_2:	37	Override to position = 0%
HVO_MINIMUM_2:	38	Override to configured minimum
HVO_MAXIMUM_2:	39	Override to configured maximum
HVO_UNUSED40:	40	
HVO_UNUSED41:	41	
HVO_UNUSED42:	42	
HVO_UNUSED43:	43	
HVO_UNUSED44:	44	

HVO_UNUSED45:	45
HVO_UNUSED46:	46
HVO_UNUSED47:	47
HVO_UNUSED48:	48

6.18 scene_t

Index: 17

SC_NUL:	-1	Invalid value
SC_RECALL:	0	Recall a specified scene.
SC_LEARN:	1	Store the current setting in the specified scene.
SC_DISPLAY:	2	Display the current scene.
SC_GROUP_OFF:	3	Report current group is off.
SC_GROUP_ON:	4	Report current group is on.
SC_STATUS_OFF:	5	Report current status is off.
SC_STATUS_ON:	6	Report current status is on.
SC_STATUS_MIXED:	7	Report current status is mixed.
SC_GROUP_STATUS:	8	Get group status.
SC_FLICK:	9	Toggle state off and then on.
SC_TIMEOUT:	10	Report a timeout occured.
SC_TIMEOUT_FLICK:	11	Report a timeout occured for a flick warning.
SC_DELAYOFF:	12	Set the state to off after a delay.
SC_DELAYOFF_FLICK:	13	Flick and then set the state to off after a delay.
SC_DELAYON:	14	Set the state to on after a delay.
SC_ENABLE_GROUP:	15	Enable the current group.
SC_DISABLE_GROUP:	16	Disable the current group.
SC_CLEANON:	17	Recall the cleaning scene.
SC_CLEANOFF:	18	Restore the previous scene.
SC_WINK:	19	Toggle to the opposite state and then restore the state.
SC_RESET:	20	Restore the factory default scene table.

SC_MODE1:	21	Manufacturer-specific mode 1.
SC_MODE2:	22	Manufacturer-specific mode 2.
SC_MODE3:	23	Manufacturer-specific mode 3.

6.19 scene_config_t

Index: 18

SCF_NUL:	-1	Invalid Value
SCF_SAVE:	0	Overwrite this scene with new data
SCF_CLEAR:	1	Delete this scene from the list
SCF_REPORT:	2	Display this scene's data
SCF_SIZE:	3	Report the number of programmed scenes
SCF_FREE:	4	Report the number of free scene storage spaces

6.20 setting_t

Index: 19

SET_NUL:	-1	Invalid value
SET_OFF:	0	Change state to off
SET_ON:	1	Change state to on, restoring the last on setting
SET_DOWN:	2	Decrease the setting by the offset supplied in the setting field
SET_UP:	3	Increase the setting by the offset supplied in the setting field
SET_STOP:	4	Stop any motion, for example for blinds
SET_STATE:	5	Change the setting to the value specified

6.21 evap_t

Index: 20

EVAP_NUL:	-1	Invalid Value
EVAP_NO_COOLING:	0	Object not performing cooling (off cycle or disabled)
EVAP_COOLING:	1	Object currently cooling
EVAP_EMERG_COOLING:	2	Object performing emergency cooling

6.22 therm_mode_t

Index: 21

THERM_NUL:	-1	Invalid Value
THERM_NO_CONTROL:	0	Thermostat disabled
THERM_IN_OUT:	1	Cut in/out control
THERM_MODULATING:	2	Modulating control

6.23 defrost_mode_t

Index: 22

DFM_NUL:	-1	Invalid Value
DFM_MODE_AMBIENT:	0	No forced heating required
DFM_MODE_FORCED:	1	Start-up after defrost ignored
DFM_MODE_SYNC:	2	Synchronized

6.24 defrost_term_t

Index: 23

DFT_NUL:	-1	Invalid Value
DFT_TERM_TEMP:	0	Terminate on temperature
DFT_TERM_TIME:	1	Terminate on time
DFT_TERM_FIRST:	2	Terminate on first occurring
DFT_TERM_LAST:	3	Terminate on last occurring
DFT_TERM_SENSOR:	4	Terminate on sensor
DFT_TERM_DISCHARGE:	5	Terminate on discharge
DFT_TERM_RETURN:	6	Terminate on return
DFT_TERM_SW_OPEN:	7	Terminate on "Switch Open"
DFT_TERM_SW_CLOSE:	8	Terminate on "Switch Closed"
DFT_TERM_MANUF:	100	Manufacturer-Defined termination state

6.25 defrost_state_t

Index: 24

DFS_NUL:	-1	Invalid Value
DFS_STANDBY:	0	Defrost in standby
DFS_PUMPDOWN:	1	Defrost in pump-down mode
DFS_DEFROST:	2	In defrost mode
DFS_DRAINDOWN:	3	Defrost in drain-down
DFS_INJECT_DLY:	4	Defrost in injection delay

6.26 chiller_t

Index: 25

CHLR_NUL:	-1	Invalid Value
CHLR_OFF:	0	Chiller off
CHLR_START:	1	Chiller in start mode
CHLR_RUN:	2	Chiller in run mode
CHLR_PRESHUTDN:	3	Chiller in pre shutdown mode
CHLR_SERVICE:	4	Chiller in service mode

6.27 fire_test_t

Index: 26

FT_NUL:	-1	Invalid Value
FT_NORMAL:	0	Return object to normal status
FT_RESET:	1	Perform a RESET function (for smoke detectors)
FT_TEST:	2	Go into TEST mode
FT_NOTEST:	3	Exit TEST mode

6.28 fire_initiator_t

Index: 27

FI_NUL:	-1	Invalid Value
FI_UNDEFINED:	0	Initiator is undefined
FI_THERMAL_FIXED:	1	Initiator is thermal fixed (heat)

FI_SMOKE_ION:	2	Initiator is smoke and ion
FI_MULTI_ION_THERMAL:	3	Initiator is multi-ion and thermal
FI_SMOKE_PHOTO:	4	Initiator is smoke and photo
FI_MULTI_PHOTO_THERMAL:	5	Initiator is multi-photo and thermal
FI_MULTI_PHOTO_ION:	6	Initiator is multi-photo and ion
FI_MULTI_PHOTO_ION_THERMAL:	7	Initiator is multi-photo, ion and thermal
FI_THERMAL_ROR:	8	Initiator is thermal fixed and Rate of Rise
FI_MULTI_THERMAL_ROR:	9	Initiator is multi-thermal and Rate of Rise
FI_MANUAL_PULL:	10	Initiator is manual pull
FI_WATER_FLOW:	11	Initiator is water flow
FI_WATER_FLOW_TAMPER:	12	Initiator is water flow and tamper
FI_STATUS_ONLY:	13	Initiator is status only
FI_MANUAL_CALL:	14	Initiator is a manual call point
FI_FIREMAN_CALL:	15	Initiator is a fireman call point
FI_UNIVERSAL:	16	General purpose initiator definition

6.29 fire_indicator_t

Index: 28

FN_NUL:	-1	Invalid Value
FN_UNDEFINED:	0	Undefined indicator
FN_STROBE_U:	1	The indicator is un-synchronized
FN_STROBE_S:	2	The indicator is synchronized
FN_HORN:	3	The indicator is a DC input, pre coded Horn
FN_CHIME:	4	The indicator is a DC input, pre coded Chime
FN_BELL:	5	The indicator is a DC input
FN_SOUNDER:	6	The indicator is powered from the device
FN_SPEAKER:	7	The indicator is an AC input for the speaker
FN_UNIVERSAL:	8	General purpose indicator

6.30 calendar_type_t

Index: 29

CAL_NUL:	-1	Invalid Value
CAL_GREG:	0	Gregorian calendar
CAL_JUL:	1	Julian calendar
CAL_MEU:	2	Calendar Method European/US "MEU"

6.31 reg_val_unit_t

Index: 30

RVU_NUL:	-1	invalid unit of measure (INVALID)
RVU_NONE:	0	no units specified ( )
RVU_W:	1	Watts (W)
RVU_KW:	2	kiloWatts (kW)
RVU_MW:	3	megaWatts (MW)
RVU_GW:	4	gigaWatts (GW)
RVU_VAR:	5	Volt-Amperes reactive (var)
RVU_KVAR:	6	kilo-Volt-Amperes reactive (kvar)
RVU_MVAR:	7	mega-Volt-Amperes reactive (Mvar)
RVU_GVAR:	8	giga-Volt-Amperes reactive (Gvar)
RVU_WH:	9	Watt-hour (Wh)
RVU_KWH:	10	kiloWatt-hour (kWh)
RVU_MWH:	11	megaWatt-hour (MWh)
RVU_GWH:	12	gigaWatt-hour (GWh)
RVU_VARH:	13	Volt-Amperes reactive -hour (varh)
RVU_KVARH:	14	kilo-Volt-Amperes reactive -hour (kvarh)
RVU_MVARH:	15	mega-Volt-Amperes reactive -hour (Mvarh)
RVU_GVARH:	16	giga-Volt-Amperes reactive -hour (Gvarh)
RVU_V:	17	Volts (V)
RVU_A:	18	Amps (A)
RVU_COSF:	19	(cosf)

RVU_M3:	20	cubic metres (m³)(cu.m)
RVU_L:	21	litres (l)
RVU_ML:	22	millilitres (ml)
RVU_USGAL:	23	U.S. Gallons (USG)
RVU_GJ:	24	giga-Joules (GJ)
RVU_MJ:	25	mega-Joules (MJ)
RVU_MCAL:	26	megacalories (Mcal)
RVU_KCAL:	27	kilocalories (kcal) / Calories (Cal)
RVU_MBTU:	28	mega-British thermal units (mBtu)
RVU_KBTU:	29	kilo-British thermal units (kBtu)
RVU_MJH:	30	mega-Joules per hour (MJ/h)
RVU_MLS:	31	millilitres per second (ml/s)
RVU_LS:	32	litres per second (l/s)
RVU_M3S:	33	cubic-metres per second (m³/s) (cu.m/s)
RVU_C:	34	(C)
RVU_LH:	35	litres per hour (l/h)
RVU_VA:	36	Volt-Amperes (VA)
RVU_KVA:	37	kiloVolt-Amperes (kVA)
RVU_MVA:	38	megaVolt-Amperes (MVA)
RVU_GVA:	39	gigaVolt-Amperes (GVA)
RVU_VAH:	40	Volt-Ampere hours (VAh)
RVU_KVAH:	41	kiloVolt-Ampere hours (kVAh)
RVU_MVAH:	42	megaVolt-Ampere hours (MVAh)
RVU_GVAH:	43	giga-Volt-Ampere hours (GVAh)

6.32 hvac_hvt_t

Index: 31

HVT_NUL:	-1	Invalid Value
HVT_GENERIC:	0	Generic
HVT_FAN_COIL:	1	Fan Coil

HVT_VAV:	2	Variable Air Volume Terminal
HVT_HEAT_PUMP:	3	Heat Pump
HVT_ROOFTOP:	4	Rooftop Unit
HVT_UNIT_VENT:	5	Unit Ventilator
HVT_CHILL_CEIL:	6	Chilled Ceiling
HVT_RADIATOR:	7	Radiator
HVT_AHU:	8	Air Handling Unit
HVT_SELF_CONT:	9	Self-Contained Unit

6.33 control_resp_t

Index: 32

CTRLR_NUL:	-1	Invalid value
CTRLR_NO:	0	Number of current controller.
CTRLR_PEND:	1	Request pending due to control query to current operator
CTRLR_REL:	2	Current control released.
CTRLR_QUERY:	3	Query to current controller.
CTRLR_RES:	4	Controllable device has been reset.
CTRLR_ERR:	5	Error in control.

6.34 pan_dir_t

Index: 33

PAN_NUL:	-1	Invalid Value
PAN_STOP:	0	Stop panning
PAN_RIGHT:	1	Pan to the right
PAN_LEFT:	2	Pan to the left

6.35 tilt_dir_t

Index: 34

TILT_NUL:	-1	Invalid Value
TILT_STOP:	0	Stop tilting
TILT_UP:	1	Tilt up
TILT_DOWN:	2	Tilt down

6.36 zoom_t

Index: 35

ZOOM_NUL:	-1	Invalid Value
ZOOM_STOP:	0	Stop zooming
ZOOM_TELE:	1	Telephoto zoom / zoom in
ZOOM_WIDE:	2	Wide zoom / zoom out

6.37 privacyzone_t

Index: 36

PZ_NUL:	-1	Invalid value
PZ_DISABLE:	0	Disable privacy zone warning
PZ_ENABLE:	1	Enable privacy zone warning
PZ_UPPER_LEFT:	2	Set upper left corner
PZ_LOWER_RIGHT:	3	Set lower right corner
PZ_ENTER:	4	Privacy zone enter warning
PZ_EXIT:	5	Privacy zone exit message

6.38 cam_func_t

Index: 37

CMF_NUL:	-1	Invalid function call response.
CMF_REL:	0	Relative positions, prepositions.
CMF_TOUR:	1	Preposition tour tables.
CMF_ABS:	2	Absolute positions.

6.39 cam_act_t

Index: 38

CMA_NUL:	-1	Invalid action call response.
CMA_SAVE:	0	Save the values defined by the function.
CMA_CALL:	1	Preposition tour tables.
CMA_READ:	2	Absolute positions.

6.40 gfci_status_t

Index: 39

GFCI_NUL:	-1	Invalid Value
GFCI_UNKNOWN:	0	Unknown response
GFCI_NORMAL:	1	Normal GFCI operating condition
GFCI_TRIPPED:	2	A ground-fault has caused the GFCI to interrupt the circuit
GFCI_TEST_FAILED:	3	The GFCI failed testing
GFCI_TEST_PASSED:	4	The GFCI passed testing
GFCI_TEST_NOW:	5	The GFCI needs to be tested

6.41 motor_state_t

Index: 40

MOTOR_NUL:	-1	The state of the motor is unknown (invalid value)
MOTOR_STOPPED:	0	The motor is not running.
MOTOR_STARTING:	1	The motor is performing its start-up sequence.
MOTOR_ACCELERATING:	2	The motor is running. Speed is increasing.
MOTOR_AT_STANDBY:	3	The motor is running in its standby mode.
MOTOR_AT_NORMAL:	4	The motor is running in its normal operational mode.
MOTOR_AT_REFERENCE:	5	The motor is running at its reference speed.
MOTOR_DECELERATING:	6	The motor is running. Speed is decreasing.
MOTOR_STOPPING:	7	The motor is running, beginning its shutdown sequence.

6.42 boolean_t

Index: 41

BOOL_NUL:	-1	Invalid Value
BOOL_FALSE:	0	False
BOOL_TRUE:	1	True

6.43 ex_control_t

Index: 42

EX_CONTROL_NUL:	-1	The control status of the item is unknown
EX_CONTROL_NONE:	0	Nothing has control of the item.
EX_CONTROL_OTHER:	1	Some unidentified entity has control of the item.
EX_CONTROL_THIS_ADDR:	2	A device has control of the item. The network address of this device is specified in the control_device_addr

6.44 unit_temp_t

Index: 43

TEMP_NUL:	-1	The status of the apparatus or unit is unknown, or not applicable (Invalid Value).
TEMP_INACTIVE:	0	The temperature-sensing apparatus is present, but not currently operating.
TEMP_AT_DESIRED:	1	The unit temperature is within the desired range.
TEMP_TOO_HOT:	2	The unit temperature is above the upper limit of the desired range.
TEMP_TOO_COLD:	3	The unit temperature is below the lower limit of the desired range.

6.45 device_c_mode_t

Index: 44

DCM_NUL:	-1	Invalid Value
DCM_SPEED_CONST:	0	
DCM_PRESS_CONST:	1	
DCM_PRESS_COMP:	2	
DCM_FLOW_CONST:	3	
DCM_FLOW_COMP:	4	
DCM_TEMP_CONST:	5	

DCM_TEMP_COMP:	6	
DCM_PRESS_AUTO:	7	
DCM_QUICK_OPEN:	20	Valve works with Quick-Open flow characteristic
DCM_LINEAR:	21	Valve works with Linear flow characteristic
DCM_EQUAL_PERCENT:	22	Valve works with Equal Percent flow characteristic
DCM_QUADRATIC:	23	Valve works with Quadratic flow characteristic
DCM_FREE_DEFINED:	24	Valve works with free defined flow characteristic
DCM_2WAY_VALVE:	27	
DCM_MIXING_VALVE:	28	
DCM_DIVERTING_VALVE:	29	
DCM_INVFNC_QCK_OPN:	30	
DCM_INVFNC_EQL_PERC:	31	
DCM_INVFNC_QUAD:	32	

6.46 valve_mode_t

Index: 45

VALVE_NUL:	-1	Invalid value
VALVE_NORMAL:	0	Valve works as normal valve
VALVE_COOLING:	1	Valve works as cooling valve only
VALVE_HEATING:	2	Valve works as heating valve only
VALVE_EMERGENCY:	3	Valve works in emergency operation
VALVE_STROKE_ADP:	4	Valve adapt its stroke and its end positions
VALVE_STROKE_SYN:	5	Valve resynchronizes its position
VALVE_ERROR:	6	Valve is in error mode
VALVE_OVERRIDDEN:	7	Value is overridden

6.47 nv_type_category_t

Index: 46

NVT_CAT_NUL:	-1	Invalid Value
NVT_CAT_INITIAL:	0	
NVT_CAT_SIGNED_CHAR:	1	8-bit signed character
NVT_CAT_UNSIGNED_CHAR:	2	8-bit unsigned character
NVT_CAT_SIGNED_SHORT:	3	8-bit signed integer
NVT_CAT_UNSIGNED_SHORT:	4	8-bit unsigned integer
NVT_CAT_SIGNED_LONG:	5	16-bit signed integer
NVT_CAT_UNSIGNED_LONG:	6	16-bit unsigned integer
NVT_CAT_ENUM:	7	8-bit enumeration
NVT_CAT_ARRAY:	8	Array
NVT_CAT_STRUCT:	9	Structure
NVT_CAT_UNION:	10	Union
NVT_CAT_BITFIELD:	11	Bitfield
NVT_CAT_FLOAT:	12	32-bit IEC 60559 (IEEE 754) floating-point value
NVT_CAT_SIGNED_QUAD:	13	32-bit signed integer
NVT_CAT_REFERENCE:	14	Reference type
NVT_CAT_UNSIGNED_QUAD:	15	32-bit unsigned integer
NVT_CAT_DOUBLE_FLOAT:	16	64-bit floating-point value
NVT_CAT_SIGNED_INT64:	17	64-bit signed integer
NVT_CAT_UNSIGNED_INT64:	18	64-bit unsigned integer

6.48 ent_opmode_cmd_t

Index: 47

EM_NUL:	-1	Invalid Value
EM_UNDEFINED:	0	Operation mode is not defined
EM_AUTO:	1	Operation mode is AUTOMATIC
EM_AUTO_RED:	2	Operation mode is AUTOMATIC with reduced width
EM_CLOSE_LOCK:	3	Operation mode is CLOSE AND LOCK

EM_CLOSE_UNLOCK:	4	Operation mode is CLOSE AND UNLOCK
EM_EXIT_ONLY:	5	Operation mode is EXIT ONLY
EM_OPEN:	6	Operation mode is OPEN
EM_OPEN_ONCE:	7	Operation mode is OPEN AND CLOSE ONCE
EM_MANUAL:	8	Operation mode is MANUAL
EM_FIRE:	9	Operation mode is FIRE
EM_EVAC:	10	Operation mode is EVACUATION
EM_WEATHER:	11	Operation mode is WEATHER MODE
EM_DAY_LOCKING:	12	Operation mode is DAY_LOCKING, locking with reduced level of security
EM_NIGHT_LOCKING:	13	Operation mode is NIGHT_LOCKING, locking with maximum level of security
EM_BLOCKED:	14	Operation mode is BLOCKED, no operations is allowed
EM_SERVICE:	15	Operation mode is SERVICE
EM_ENTRY_ONLY:	16	Operation mode is ENTRY_ONLY

6.49 ent_cmd_t

Index: 48

ES_NUL:	-1	Invalid Value
ES_UNDEFINED:	0	State is not yet defined
ES_OPEN_PULS:	1	Open the device and close it when back in normal position
ES_OPEN:	2	Open the device if not locked
ES_CLOSE:	3	Close the device
ES_STOP:	4	Stop the device
ES_STOP_RESUME:	5	Continue after stop command
ES_ENTRY_REQ:	6	Entry request, access in to the area
ES_EXIT_REQ:	7	Exit request, access out from the area
ES_KEY_REQ:	8	Key request, open the device if not in EM_BLOCKED or EM_MANUAL mode
ES_SAFETY_EXT_REQ:	9	Safety request, the device will go to a pre-defined safety position/mode
ES_EMERGENCY_REQ:	10	Emergency request, the device will go to an pre-defined emergency position/mode

ES_UPDATE_STATE:	11	Update the current state and mode
ES_SAF_EXT_RESUME:	12	Resume after Safety function
ES_EMERG_RESUME:	13	Resume after Emergency function

6.50 flow_direction_t

Index: 49

FD_NUL:	-1	Invalid Value
FD_NONE:	0	No flow/movement allowed
FD_OUT:	1	Exit/out/away direction only
FD_IN:	2	Entry/in/toward direction only
FD_ANY:	3	No restriction on flow/movement

6.51 device_select_t

Index: 50

DV_NUL:	-1	Invalid value
DV_PUMP_CTRL:	0	Use union for SFPTpumpController values
DV_VALVE_POS:	1	Use union for SFPTvalvePositioner values

6.52 event_mode_type_t

Index: 51

EMT_NUL:	-1	Invalid Value
EMT_END_OF_LIST:	0	End of list indicator
EMT_SCENE:	1	Scene indicator
EMT_MODE:	2	Mode indicator
EMT_LIGHTS_ON:	3	
EMT_LIGHTS_OFF:	4	

6.53 master_slave_t

Index: 52

MSC_NUL:	-1	Invalid Value
MSC_UNKNOWN:	0	Undefined or unused
MSC_SLAVE:	1	Slave control
MSC_MASTER:	2	Master control

6.54 fan_operation_t

Index: 53

HVF_NUL:	-1	Invalid Value
HVF_CONTINUOUS:	0	Fan runs continuously
HVF_CYCLE:	1	Fan cycles with heating and cooling
HVF_CON_CYCLE:	2	Continuous in occupied, cycles in occupied stand by
HVF_CYCLE_HEAT:	3	Fan cycles with heating only
HVF_CYCLE_COOL:	4	Fan cycles with cooling only

6.55 days_of_month_t

Index: 54

DM_NUL:	-1	Invalid value
DM_EVERY_DAY:	0	Every day of month
DM_DAY_1:	1	First day of month
DM_DAY_2:	2	Second day of month
DM_DAY_3:	3	Third day of month
DM_DAY_4:	4	Fourth day of month
DM_DAY_5:	5	Fifth day of month
DM_DAY_6:	6	Sixth day of month
DM_DAY_7:	7	Seventh day of month
DM_DAY_8:	8	Eighth day of month
DM_DAY_9:	9	Ninth day of month
DM_DAY_10:	10	Tenth day of month

DM_DAY_11:	11	Eleventh day of month
DM_DAY_12:	12	Twelfth day of month
DM_DAY_13:	13	Thirteenth day of month
DM_DAY_14:	14	Fourteenth day of month
DM_DAY_15:	15	Fifteenth day of month
DM_DAY_16:	16	Sixteenth day of month
DM_DAY_17:	17	Seventeeth day of month
DM_DAY_18:	18	Eighteenth day of month
DM_DAY_19:	19	Nineteenth day of month
DM_DAY_20:	20	Twentieth day of month
DM_DAY_21:	21	Twenty-first day of month
DM_DAY_22:	22	Twenty-second day of month
DM_DAY_23:	23	Twenty-third day of month
DM_DAY_24:	24	Twenty-fourth day of month
DM_DAY_25:	25	Twenty-fifth day of month
DM_DAY_26:	26	Twenty-sixth day of month
DM_DAY_27:	27	Twenty-seventh day of month
DM_DAY_28:	28	Twenty-eighth day of month
DM_DAY_29:	29	Twenty-ninth day of month
DM_DAY_30:	30	Thirtieth day of month
DM_DAY_31:	31	Thirty-first day of month
DM_LAST_DAY_OF_MONTH:	32	Last day of month
DM_LAST_SECOND_DAY:	33	Second to last day of month
DM_LAST_THIRD_DAY:	34	Third to last day of month
DM_LAST_4TH_DAY:	35	Fourth to last day of month
DM_LAST_5TH_DAY:	36	Fifth to last day of month
DM_LAST_6TH_DAY:	37	Sixth to last day of month
DM_LAST_7TH_DAY:	38	Seventh to last day of month
DM_LAST_8TH_DAY:	39	Eighth to last day of month

DM_LAST_9TH_DAY:	40	Ninth to last day of month
DM_LAST_10TH_DAY:	41	Tenth to last day of month
DM_LAST_11TH_DAY:	42	Eleventh to last day of month
DM_LAST_12TH_DAY:	43	Twelfth to last day of month
DM_LAST_13TH_DAY:	44	Thirteenth to last day of month
DM_LAST_14TH_DAY:	45	Fourteenth to last day of month
DM_LAST_15TH_DAY:	46	Fifteenth to last day of month
DM_LAST_16TH_DAY:	47	Sixteenth to last day of month
DM_LAST_17TH_DAY:	48	Seventeenth to last day of month
DM_LAST_18TH_DAY:	49	Eighteenth to last day of month
DM_LAST_19TH_DAY:	50	Nineteenth to last day of month
DM_LAST_20TH_DAY:	51	Twentieth to last day of month
DM_LAST_21ST_DAY:	52	Twenty-first to last day of month
DM_LAST_22ND_DAY:	53	Twenty-second to last day of month
DM_LAST_23RD_DAY:	54	Twenty-third to last day of month
DM_LAST_24TH_DAY:	55	Twenty-fourth to last day of month
DM_LAST_25TH_DAY:	56	Twenty-fifth to last day of month
DM_LAST_26TH_DAY:	57	Twenty-sixth to last day of month
DM_LAST_27TH_DAY:	58	Twenty-seventh to last day of month
DM_LAST_28TH_DAY:	59	Twenty-eighth to last day of month
DM_LAST_29TH_DAY:	60	Twenty-ninth to last day of month
DM_LAST_30TH_DAY:	61	Thirtieth to last day of month
DM_FIRST_SUN:	62	First Sunday of month
DM_FIRST_MON:	63	First Monday of month
DM_FIRST_TUE:	64	First Tuesday of month
DM_FIRST_WED:	65	First Wednesday of month
DM_FIRST_THU:	66	First Thursday of month
DM_FIRST_FRI:	67	First Friday of month
DM_FIRST_SAT:	68	First Saturday of month

DM_SECOND_SUN:	69	Second Sunday of month
DM_SECOND_MON:	70	Second Monday of month
DM_SECOND_TUE:	71	Second Tuesday of month
DM_SECOND_WED:	72	Second Wednesday of month
DM_SECOND_THU:	73	Second Thursday of month
DM_SECOND_FRI:	74	Second Friday of month
DM_SECOND_SAT:	75	Second Saturday of month
DM_THIRD_SUN:	76	Third Sunday of month
DM_THIRD_MON:	77	Third Monday of month
DM_THIRD_TUE:	78	Third Tuesday of month
DM_THIRD_WED:	79	Third Wednesday of month
DM_THIRD_THU:	80	Third Thursday of month
DM_THIRD_FRI:	81	Third Friday of month
DM_THIRD_SAT:	82	Third Saturday of month
DM_FOURTH_SUN:	83	Fourth Sunday of month
DM_FOURTH_MON:	84	Fourth Monday of month
DM_FOURTH_TUE:	85	Fourth Tuesday of month
DM_FOURTH_WED:	86	Fourth Wednesday of month
DM_FOURTH_THU:	87	Fourth Thursday of month
DM_FOURTH_FRI:	88	Fourth Friday of month
DM_FOURTH_SAT:	89	Fourth Saturday of month
DM_FIFTH_SUN:	90	Fifth Sunday of month
DM_FIFTH_MON:	91	Fifth Monday of month
DM_FIFTH_TUE:	92	Fifth Tuesday of month
DM_FIFTH_WED:	93	Fifth Wednesday of month
DM_FIFTH_THU:	94	Fifth Thursday of month
DM_FIFTH_FRI:	95	Fifth Friday of month
DM_FIFTH_SAT:	96	Fifth Saturday of month
DM_LAST_SUN:	97	Last Sunday of month

DM_LAST_MON:	98	Last Monday of month
DM_LAST_TUE:	99	Last Tuesday of month
DM_LAST_WED:	100	Last Wednesday of month
DM_LAST_THU:	101	Last Thursday of month
DM_LAST_FRI:	102	Last Friday of month
DM_LAST_SAT:	103	Last Saturday of month
DM_EVERY_SUN:	104	Every Sunday of the month
DM_EVERY_MON:	105	Every Monday of the month
DM_EVERY_TUE:	106	Every Tuesday of the month
DM_EVERY_WED:	107	Every Wednesday of the month
DM_EVERY_THU:	108	Every Thursday of the month
DM_EVERY_FRI:	109	Every Friday of the month
DM_EVERY_SAT:	110	Every Saturday of the month
DM_EVERY_SECOND_DAY:	111	Every second day (i.e. every other day) of the date
DM_EVERY_THIRD_DAY:	112	Every third day of the date interval
DM_EVERY_FOURTH_DAY:	113	Every fourth day of the date interval
DM_EVERY_FIFTH_DAY:	114	Every fifth day of the date interval
DM_EVERY_SIXTH_DAY:	115	Every sixth day of the date interval
DM_EVERY_WEEKDAY:	116	Every weekday (Monday - Friday)
DM_EVERY_WEEKEND_DAY:	117	Every weekend day (Saturday - Sunday)

6.56 months_t

Index: 55

MN_NUL:	-1	Invalid value
MN_EVERY_MONTH:	0	Every month
MN_JAN:	1	January
MN_FEB:	2	February
MN_MAR:	3	March
MN_APR:	4	April
MN_MAY:	5	May

MN_JUN:	6	June
MN_JUL:	7	July
MN_AUG:	8	August
MN_SEP:	9	September
MN_OCT:	10	October
MN_NOV:	11	November
MN_DEC:	12	December
MN_EVERY_2_MONTH:	13	Every other month
MN_QUARTERLY:	14	Every third month
MN_EVERY_4_MONTH:	15	Every fourth month
MN_EVERY_5_MONTH:	16	Every fifth month
MN_EVERY_6_MONTH:	17	Every sixth month
MN_EVERY_7_MONTH:	18	Every seventh month
MN_EVERY_8_MONTH:	19	Every eighth month
MN_EVERY_9_MONTH:	20	Every ninth month
MN_EVERY_10_MONTH:	21	Every tenth month
MN_EVERY_11_MONTH:	22	Every eleventh month
MN_EVERY_ODD_MONTH:	23	Jan, Mar, May, Jul, Sep, Nov
MN_EVERY_EVEN_MONTH:	24	Feb, Apr, Jun, Aug, Oct, Dec

6.57 sec_status_t

Index: 56

SSS_NUL:	-1
SSS_POWER_UP:	0
SSS_ALARM_RESET:	1
SSS_ALARM:	2
SSS_TAMPER_RESET:	3
SSS_TAMPER:	4
SSS_MAINTENANCE:	5

SSS_TROUBLE: 6

SSS_FAULT: 7

SSS_RECOVERED_SENSOR: 8

SSS_LOST_SENSOR: 9

SSS_POLL_ACTIVE: 10

SSS_POLL_INACTIVE: 11

SSS_POLL_TAMPER: 12

SSS_POLL_ON: 13

SSS_POLL_OFF: 14

SSS_POLL_INHIBIT: 15

SSS_POLL_TEST: 16

SSS_CONFIRM_OFF: 17

SSS_CONFIRM_ON: 18

SSS_CONFIRM_INHIBIT_RESET: 19

SSS_CONFIRM_INHIBIT: 20

SSS_CONFIRM_WALK_TEST_OFF: 21

SSS_CONFIRM_WALK_TEST_ON: 22

SSS_CONFIRM_TEST_MODE_OFF: 23

SSS_CONFIRM_TEST_MODE_ON: 24

SSS_CONFIRM_UNSUPPORTED: 25

6.58 sec_state_t

Index: 57

SSE_NUL: -1

SSE_OFF: 0

SSE_ON: 1

SSE_INHIBIT_RESET: 2

SSE_INHIBIT:	3
SSE_WALK_TEST_OFF:	4
SSE_WALK_TEST_ON:	5
SSE_TEST_MODE_OFF:	6
SSE_TEST_MODE_ON:	7
SSE_POLL_STATUS:	8
SSE_POLL_STATE:	9
SSE_CONFIRM_ALARM_RESET:	10
SSE_CONFIRM_ALARM:	11
SSE_CONFIRM_TAMPER_RESET:	12
SSE_CONFIRM_TAMPER:	13
SSE_CONFIRM_MAINTENANCE:	14
SSE_CONFIRM_TROUBLE:	15
SSE_CONFIRM_FAULT:	16
SSE_CONFIRM_RECOVERED_SENSOR:	17
SSE_LOST_SENSOR:	18
SSE_CONFIRM_UNSUPPORTED:	19

6.59 interval_of_month_t

Index: 58

IOM_NUL:	-1	Invalid Value
IOM_MINUTE:	0	Interval in minutes
IOM_HOUR:	1	Interval in hours
IOM_DAY:	2	Interval in days
IOM_WEEK:	3	Interval in weeks
IOM_MONTH:	4	Interval in months

6.60 sblnd_cmd_source_t

Index: 59

SBCS_NUL:	-1	Invalid value
SBCS_LOCAL:	0	Local
SBCS_GROUP:	1	Group
SBCS_WIND_SPEED:	2	Wind speed
SBCS_SUN_LUX:	3	Sun lux level
SBCS_RAIN:	4	Rain
SBCS_FROST:	5	Frost
SBCS_DAWN:	6	Dawn
SBCS_DUSK:	7	Dusk
SBCS_OUTSIDE_TEMP:	8	Outside temperature
SBCS_INDOOR_TEMP:	9	Indoor temperature
SBCS_OUTDOOR_RH:	10	Outdoor relative humidity
SBCS_INDOOR_RH:	11	Indoor relative humidity
SBCS_ILLUM_LEVEL:	12	Illumination level
SBCS_SCENE:	13	Scene
SBCS_GLOBAL:	14	Global
SBCS_WINDOW_CONTACT:	15	Window contact
SBCS_AUTOMODE_CHANGED:	16	Auto-mode changed
SBCS_OVERRIDE:	17	Override
SBCS_EMERGENCY:	18	Emergency
SBCS_MAINTENANCE:	19	Maintenance
SBCS_INTRUSION:	20	Intrusion
SBCS_TERMINAL_LOAD:	21	Terminal load
SBCS_ALARM:	22	Alarm
SBCS_OCC_SENSOR:	23	Occupancy sensor
SBCS_OCC_MAN_CMD:	24	Occupancy manual command
SBCS_GLARE:	25	Glare

SBCS_ALARM_2:	26	Alarm 2
SBCS_NOTIFY:	27	Notify
SBCS_ELEVATION:	28	Elevation
SBCS_AZIMUTH:	29	Azimuth
SBCS_SET_OVERRIDE:	30	Set override
SBCS_SET_MAINTENANCE:	31	Set maintenance
SBCS_TIMER:	32	Timer
SBCS_UNKNOWN:	127	Unknown command source

6.61 sbInd_error_t

Index: 60

SBE_NUL:	-1	Invalid Value
SBE_NO_ERROR:	0	No error
SBE_IN_PROGRESS:	1	In progress
SBE_LIMITS:	2	Limits
SBE_OBSTACLE_UP:	3	Obstacle up
SBE_OBSTACLE_DOWN:	4	Obstacle down
SBE_OVERHEAT:	5	Overheat
SBE_POWER:	6	Power
SBE_SENSOR:	7	Sensor
SBE_MOTOR_CIRCUIT:	8	Motor circuit
SBE_FUSE:	9	Fuse
SBE_REFERENCE_LOST:	10	Reference lost
SBE_HOST_COMM:	11	Host communication
SBE_VOLTAGE_1:	12	Voltage 1
SBE_VOLTAGE_2:	13	Voltage 2
SBE_CONTROLLER:	14	Controller

6.62 rail_audio_sensor_type_t

Index: 61

RAST_NUL:	-1	Invalid Value
RAST_CU_TYPE_1:	0	CU Type 1
RAST_CU_TYPE_2:	1	CU Type 2
RAST_CU_TYPE_3:	2	CU Type 3
RAST_CU_TYPE_4:	3	CU Type 4
RAST_LS_LINE_1:	4	LS Line 1
RAST_LS_LINE_2:	5	LS Line 2
RAST_LS_LINE_3:	6	LS Line 3
RAST_LS_LINE_4:	7	LS Line 4
RAST_LS_LINE_5:	8	LS Line 5
RAST_LS_LINE_6:	9	LS Line 6
RAST_LS_LINE_7:	10	LS Line 7
RAST_LS_LINE_8:	11	LS Line 8
RAST_PAU:	12	Public-Address Unit
RAST_CFA_TYPE_1:	13	CFA Type 1
RAST_CFA_TYPE_2:	14	CFA Type 2
RAST_CFA_TYPE_3:	15	CFA Type 3
RAST_CFA_TYPE_4:	16	CFA Type 4
RAST_DVA:	17	DVA
RAST_ET_TYPE_1:	18	ET Type 1
RAST_ET_TYPE_2:	19	ET Type 2
RAST_USERDEF_TYPE_1:	20	User-defined Type 1
RAST_USERDEF_TYPE_2:	21	User-defined Type 2
RAST_USERDEF_TYPE_3:	22	User-defined Type 3
RAST_USERDEF_TYPE_4:	23	User-defined Type 4

6.63 rail_audio_type_t

Index: 62

RAT_NUL:	-1
RAT_IC_REQ:	0
RAT_IC_JOIN:	1
RAT_IC_QUIT:	2
RAT_IC_END:	3
RAT_HW_RADIO_REQ:	4
RAT_HW_RADIO_END:	5
RAT_HW_PA_REQ:	6
RAT_HW_PA_END:	7
RAT_SW_PA_REQ:	8
RAT_SW_PA_END:	9
RAT_SW_PA_OR_REQ:	10
RAT_SW_PA_OR_END:	11
RAT_PAU_REQ:	12
RAT_PAU_ACCEPT:	13
RAT_PAU_CALL:	14
RAT_PAU_END:	15
RAT_ENTERT_REQ:	16
RAT_ENTERT_END:	17

6.64 appl_cwc_t

Index: 63

CWC_NUL:	-1	Invalid Value
CWC_WASH:	0	Wash
CWC_RINSE:	1	Rinse
CWC_SPIN:	2	Spin
CWC_DRY:	3	Dry

6.65 appl_cws_t

Index: 64

CWS_NUL:	-1	Invalid Value
CWS_LOAD_SENSING:	0	Sensing Load
CWS_WETTING:	1	Wetting
CWS_DETERGENT:	2	Detergent
CWS_WASHING:	3	Washing
CWS_WATERING:	4	Watering
CWS_RINSING:	5	Rinsing
CWS_ARRANGING:	6	Arranging
CWS_DRAIN:	7	Drain
CWS_SPINNING:	8	Spinning
CWS_FINAL_SPINNING:	9	In Final Spin
CWS_FLUFFING:	10	Fluffing
CWS_DRYING:	11	Drying
CWS_COOLING:	12	Cooling

6.66 appl_cwp_t

Index: 65

CWP_NUL:	-1	Invalid Value
CWP_GENERAL:	0	Normal Wash
CWP_BOIL:	1	Boil

CWP_FAST_WASH:	2	Fast Wash
CWP_LINGERIE:	3	Lingerie
CWP_WOOL:	4	Wool
CWP_TOWEL:	5	Towel
CWP_BED_LINENS:	6	Bed Linens
CWP_CURTAIN:	7	Curtain
CWP_RINSE_SPIN_ONLY:	8	Rinse and Spin Only
CWP_DELICATE_RINSE:	9	Delicate Rinse
CWP_SPIN_ONLY:	10	Spin Only
CWP_DRY_ONLY:	11	Dry Only

6.67 appl_rin_t

Index: 66

RIN_NUL:	-1	Invalid Value
RIN_PRE_WASH:	0	Pre-wash
RIN_WATER_PLUS:	1	Water Plus
RIN_DETERGENT_PLUS:	2	Detergent Plus
RIN_RINSE_HOLD:	3	Rinse Hold

6.68 aham_appl_t

Index: 67

AHAM_NUL:	-1	Invalid Value
AHAM_CLOTHES_WASHER:	0	Clothes Washer
AHAM_REFRIGERATOR_FREEZER:	1	Refrigerator Freezer
AHAM_CLOTHES_DRYER:	2	Clothes Dryer
AHAM_DISHWASHER:	3	Dishwasher
AHAM_RANGE_OVEN_COOKTOP:	4	Range Oven Cooktop
AHAM_COUNTERTOP_MICROWAVE_OVEN:	5	Countertop Microwave Oven
AHAM_ROOM_AIR_CONDITIONER:	6	Room Air Conditioner

6.69 button_action_t

Index: 68

BTA_NUL:	-1	Invalid value
BTA_TOGGLE_STATE:	0	Toggle on-off state; same action as SW_SET_OFF if the on/off state was on, and SW_SET_ON if the on/off state was off; ignored for blinds, drapes, shades, and fans
BTA_TOGGLE_SCENE:	1	Toggle on-off state if specified scene is the current scene; recall the state from the specified scene if the scene is new
BTA_SET_STATE_ON:	2	Set the state to on; ignored for blinds, drapes, shades, and fans
BTA_RECALL_SCENE:	3	Recall a scene
BTA_SET_STATE_OFF:	4	Set the state to off; ignored for blinds, drapes, shades, and fans
BTA_SET_OCCUPIED:	5	Set the occupancy state
BTA_CLEAR_OCCUPIED:	6	Clear the occupancy state
BTA_SET_UNOCCUPIED:	7	Set the unoccupied state
BTA_CLEAR_UNOCCUPIED:	8	Clear the unoccupied state
BTA_SET_LEVEL:	9	Set the level to the specified value; ignored for blinds, drapes, shades, and fans
BTA_SET_UP_DIRECTION:	10	Set ceiling fan direction to up, with specified level
BTA_SET_DOWN_DIRECTION:	11	Set ceiling fan direction to down, with specified level
BTA_INCREASE:	12	Increase the level by specified amount; ignored for blinds, drapes, shades, and fans
BTA_DECREASE:	13	Decrease the level by the specified amount; ignored for blinds, drapes, shades, and fans
BTA_CYCLE:	14	Same as increase until 100 % is reached, then same as decrease until 0 % is reached, then repeat; ignored for blinds, drapes, shades, and fans;
BTA_ROTATE_OPEN:	15	Rotate blinds open by the setting
BTA_ROTATE_CLOSED:	16	Rotate blinds closed by the setting
BTA_SET_ANGLE:	17	Set the rotation angle of blinds to the setting
BTA_TOGGLE_DIRECTION:	18	Toggle ceiling fan direction, with specified level
BTA_TOGGLE_OCCUPANCY:	19	Toggle the occupancy state
BTA_LEARN_SCENE:	20	Learn a scene from current settings
BTA_SET_STANDBY:	21	Set standby mode

BTA_CLEAR_STANDBY:	22	Clear standby mode
BTA_TOGGLE_STANDBY:	23	Toggle standby mode
BTA_SET_FAN_ON:	24	Set the fan state to on
BTA_SET_FAN_OFF:	25	Set the fan state to off
BTA_TOGGLE_FAN_STATE:	26	Toggle the fan on-off state
BTA_INCREASE_FAN_LEVEL:	27	Increase fan speed by the specified amount
BTA_DECREASE_FAN_LEVEL:	28	Decrease fan speed by the specified amount
BTA_CYCLE_FAN_LEVEL:	29	Increase fan speed by the specified amount until the level reaches 100 %, then decrease the fan speed by the specified amount
BTA_MOVE_OPEN:	30	Move blinds, drapes, or shades open by the specified amount
BTA_MOVE_CLOSED:	31	Move blinds, drapes, or shades open by the specified amount
BTA_SET_POSITION:	32	Set blinds, drapes, or shades to the specified position; 100 % is fully open, 0 % is fully closed
BTA_STOP:	33	Stop any motion of blinds, drapes, or shades
BTA_TOGGLE_GROUP:	34	Toggle group state
BTA_ENABLE_GROUP:	35	Enable a group; all groups are enabled by default
BTA_DISABLE_GROUP:	36	Disable a group
BTA_INCREASE_HUE:	37	Increase hue
BTA_DECREASE_HUE:	38	Decrease hue
BTA_SET_DR_EVENT:	39	Set demand-response mode
BTA_CLEAR_DR_EVENT:	40	Clear demand-response mode
BTA_TOGGLE_DR_EVENT:	41	Toggle demand-response mode

6.70 char_encoding_t

Index: 69

CE_NUL:	-1	Invalid value
CE_UTF_8:	0	UTF-8 encoding
CE_UTF_16:	1	UTF-16 encoding
CE_GB18030:	2	GB18030 encoding

6.71 switch_state_t

Index: 71

SW_NUL:	-1	Invalid value
SW_SET_OFF:	0	Set the state to off; ignored for blinds, drapes, shades, and fans
SW_SET_ON:	1	Set the state to on; ignored for blinds, drapes, shades, and fans
SW_REPORT_OFF:	2	Report that the state is off; output only; ignored for input
SW_REPORT_ON:	3	Report that the state is on; output only; ignored for input
SW_TOGGLE_STATE:	4	Toggle on-off state; same action as SW_SET_OFF if the on/off state was on, and SW_SET_ON if the on/off state was off; ignored for blinds, drapes, shades, and fans
SW_SET_LEVEL:	5	Set the level to the specified value; ignored for blinds, drapes, shades, and fans
SW_INCREASE_LEVEL:	6	Set the level to the specified value; ignored for blinds, drapes, shades, and fans
SW_DECREASE_LEVEL:	7	Decrease the level by the specified amount; ignored for blinds, drapes, shades, and fans
SW_RECALL_SCENE:	8	Recall the state and level from the specified scene
SW_STORE_SCENE:	9	Store setting for the specified scene
SW_LEARN_SCENE:	10	Learn setting for the specified scene
SW_SET_OCCUPIED:	11	Set the occupancy state
SW_SET_UNOCCUPIED:	12	Clear the occupancy state
SW_SET_MULTIPLIER:	13	Set a multiplier for the level for 60 min; ignored for blinds, drapes, shades, and fans
SW_ENABLE_GROUP:	14	Enable a group; all groups are enabled by default
SW_DISABLE_GROUP:	15	Disable a group
SW_WINK:	16	Blink state (toggle on-off state; pause; toggle on-off state again
SW_RESET:	17	Reset scene definitions, multiplier, occupancy state, group enable flags, and settings to factory defaults
SW_RESET_ENERGY_USAGE:	18	Reset energy usage value to zero
SW_RESET_RUNTIME:	19	Reset runtime value to zero
SW_INCREASE_HUE:	20	Increase colour hue

SW_DECREASE_HUE:	21	Decrease colour hue
SW_SET_BUTTON:	22	Trigger the actions for pressing and releasing the button specified in the value field
SW_SET_GROUP_STATE_LEVEL:	23	Set state and percent of full level (value field) for a group specified in the scene field
SW_SET_FAN_UP:	32	Set ceiling fan direction to up, with specified level
SW_SET_FAN_DOWN:	33	Set ceiling fan direction to down, with specified level
SW_TOGGLE_FAN_DIRECTION:	34	Toggle fan up-down direction
SW_INCREASE_FAN_LEVEL:	35	Increase fan speed by the setting
SW_DECREASE_FAN_LEVEL:	36	Decrease fan speed by the setting
SW_SET_FAN_ON:	37	Set the fan state to on
SW_SET_FAN_OFF:	38	Set the fan state to off
SW_TOGGLE_FAN_STATE:	39	Toggle the fan on-off state
SW_MOVE_OPEN:	48	Move blinds, drapes, or shades open by the setting
SW_MOVE_CLOSED:	49	Move blinds, drapes, or shades closed by the setting
SW_SET_ANGLE:	50	Set the rotation angle of blinds to the setting
SW_ROTATE_OPEN:	51	Rotate blinds open by the setting
SW_ROTATE_CLOSED:	52	Rotate blinds closed by the setting
SW_STOP:	53	Stop any motion of blinds, drapes, or shades
SW_SET_STANDBY:	54	Set Standby mode
SW_TOGGLE_STANDBY:	55	Toggle the standby state
SW_SET_POSITION:	56	Set blinds, drapes, or shades to the specified position; 100 % is fully open, 0 % is fully closed
SW_REPORT_POSITION:	57	Report the position of blinds, drapes, or shades output only; ignored for input
SW_REPORT_FAN_LEVEL:	58	Report the fan speed in percent of full level output only; ignored for input

6.72 color_encoding_t

Index: 72

COLOR_NUL:	-1	Invalid value
COLOR_CIE31_LUMEN:	0	CIE31 colour space; Y output in lumen
COLOR_CIE31_PERCENT:	1	CIE31 colour space; Y output in percent of maximum lumen output of the lamp
COLOR_RGB:	2	No colour space, RGB colour value
COLOR_TEMPERATURE:	3	Colour temperature

6.73 log_status_t

Index: 73

LS_NUL:	-1	Invalid value
LS_ENABLED:	0	Log enabled
LS_DISABLED:	1	Log disabled
LS_FULL:	2	Log enabled and full
LS_OVERFLOW_ERR:	3	Log enabled, overflow occurred
LS_INVALID_LOG_ERR:	4	Invalid log selected
LS_APP_ERR:	5	Other application error

6.74 log_type_t

Index: 74

LT_NUL:	-1	Invalid value
LT_CIRCULAR:	0	Discard oldest data when full
LT_HISTORICAL:	1	Stop logging when full
LT_SNAPSHOT:	2	Only maintain the current value of each data point

6.75 timestamp_t

Index: 75

TS_NUL:	-1	Invalid value
TS_FULL:	0	Full timestamp
TS_OFFSET:	1	Offset since last full timestamp
TS_NONE:	2	No timestamp

6.76 log_record_t

Index: 76

LR_NUL:	-1	Invalid value
LR_DATA:	0	Point value
LR_LOG_STATUS:	1	Log status change
LR_TIME_CHANGE:	2	Time change

6.77 point_status_t

Index: 77

PS_NUL:	-1	Invalid value
PS_NORMAL:	0	Normal state
PS_IN_ALARM:	1	In alarm
PS_FAULT:	2	Fault not indicated by an alarm
PS_OVERRIDDEN:	3	Point value overridden
PS_OUT_OF_SERVICE:	4	Out of service

6.78 message_code_t

Index: 78

MC_NUL:	-1	Invalid value; this enumeration is not currently used in any SNVTs or SCPTs
MC_FIRST_RESERVED_CODE:	48	First reserved standard message code. Codes 48 - 62 are reserved for standard message codes.
MC_BROADCAST:	59	Broadcast message.
MC_DATA_LOG_ACCESS:	60	Data log access request. See log_response_code_t for response codes.
MC_ISI:	61	Interoperable self-installation (ISI) message
MC_FILE_TRANSFER:	62	File transfer message

6.79 log_access_req_t

Index: 79

LAR_NUL:	-1	Invalid value; this enumeration is not currently used in any SNVTs or SCPTs
LAR_GET_FIRST:	0	Get first record of a data log.
LAR_GET_NEXT:	1	Get next record of a data log.
LAR_CLEAR:	2	Clear data log.

6.80 log_response_code_t

Index: 80

LRC_NUL:	-1	Invalid value; this enumeration is not currently used in any SNVTs or SCPTs
LRC_SUCCESS:	48	The operation was successful. The payload is the requested record.
LRC_END_OF_LOG:	49	The end of the log has been reached. No payload.
LRC_VER_MISMATCH:	50	Protocol version mismatch. The payload is the supported version number closest to the requested version number.
LRC_BAD_REQUEST:	51	Unknown request type. No payload.
LRC_BAD_LOG_INDEX:	52	Index is out of range. Payload will contain the number of logs in the device.

6.81 address_type_t

Index: 81

DA_NUL:	-1	Invalid value; this enumeration is not currently used in any SNVTs or SCPTs
DA_SN:	1	Device address as subnet/node address
DA_NI:	2	Device address as unique node ID address

6.82 olc_select_t

Index: 82

OLC_NUL:	-1	Invalid value
OLC_DEFAULT:	0	Standard (default)
OLC_RELAY:	1	Relay Actuation
OLC_ECO_MODE:	2	ECO Mode
OLC_1_to_10:	3	1-to-10 Volt
OLC_DALI:	4	DALI

6.83 program_state_t

Index: 83

PRS_NUL:	-1	Invalid Value
PRS_NO_PROGRAM:	0	No Program
PRS_IDLE:	1	Idle (ready to run)
PRS_LOADING:	2	Loading (program is being loaded - will become Idle when done)
PRS_RUNNING:	3	Running (may be halted by user, or halted if error occurs)
PRS_HALTED:	4	Halted (program has stopped due to an error or user command)
PRS_UNLOADING:	5	Unloading (program is being unloaded - will become "No Program" when done)

6.84 file_type_t

Index: 84

FILE_NUL:	-1	Invalid value
FILE_VALUE:	1	LW-FTP value file
FILE_TEMPLATE:	2	LW-FTP template file
FILE_DATALOG:	3	Data log file
FILE_PROGRAM:	4	Application program file

6.85 program_status_error_t

Index: 85

PSE_NUL:	-1	Invalid value
PSE_NO_ERROR:	0	No Error
PSE_PROGRAM_FAULT_NOHALT:	1	Program fault (no halt)
PSE_INVALID_OPERATION_NOHALT:	2	Invalid operation (no halt)
PSE_INVALID_PARAMETER_NOHALT:	3	Invalid parameter (no halt)
PSE_STACK_OVERFLOW_NOHALT:	4	Stack overflow (no halt)
PSE_STACK_UNDERFLOW_NOHALT:	5	Stack underflow (no halt)
PSE_INSUFFICIENT_MEMORY_NOHALT:	6	Insufficient memory (no halt)
PSE_WATCHDOG_NOHALT:	7	Unknown error (resulted in a program halt)
PSE_UNKNOWN_ERROR_NOHALT:	31	Unknown error (no halt)
PSE_LOAD_ERROR_HALT:	32	Load error
PSE_PROGRAM_FAULT_HALT:	33	Program fault (resulted in a program halt)
PSE_INVALID_OPERATION_HALT:	34	Invalid operation (resulted in a program halt)
PSE_INVALID_PARAMETER_HALT:	35	Invalid operation (resulted in a program halt)
PSE_STACK_OVERFLOW_HALT:	36	Invalid operation (resulted in a program halt)
PSE_STACK_UNDERFLOW_HALT:	37	Stack underflow (resulted in a program halt)
PSE_INSUFFICIENT_MEMORY_HALT:	38	Insufficient Memory Halt
PSE_WATCHDOG_HALT:	39	Watchdog Halt
PSE_CORRUPTED_PROGRAM_HALT:	40	Corrupted program (resulted in a program halt)
PSE_UNKNOWN_ERROR_HALT:	63	Unknown error (resulted in a program halt)

6.86 time_source_t

Index: 86

TMS_NUL:	-1	Invalid value
TMS_SCHEDULER_NV:	0	Time source is scheduler NV input
TMS_NODE_OBJECT_NV:	1	Time source is Node Object NV input
TMS_SE2_TIME_CLIENT_NV:	2	Time source is Smart Energy 2.0 Time Client NV input
TMS_HARDWARE:	3	Time source is local hardware real time clock
TMS_ALTERNATE:	4	Alternate time source such as an SNTP server

6.87 scheduler_status_t

Index: 87

SCH_NUL:	-1	Invalid value
SCH_DAILY_SCHEDULE:	0	Daily schedule (lowest priority)
SCH_SCHED_SPECIAL:	1	Scheduled vacation or holiday event
SCH_SCHED_EXCEPTION:	2	Scheduled exception event
SCH_LOCAL_OCC_OVERRIDE:	3	Local occupancy override
SCH_EXCEPTION_SCH_OVERRIDE:	4	Exception schedule override
SCH_MANUAL_OVERRIDE:	5	Manual override (highest priority)
SCH_OTHER:	6	Undefined override

7 Standard functional profiles

7.1 General

A device application is divided into one or more *functional blocks*. A functional block is a portion of a device's application that performs a task by receiving configuration and operational data inputs, processing the input data, and sending operational data outputs. A functional block may receive inputs from the network, hardware attached to the device, or from other functional blocks on a device. A functional block may send outputs to the network, to hardware attached to the device, or to other functional blocks on the device.

The device application implements a functional block for each function on the device to which other devices should communicate, or that requires configuration for particular application behaviour. Each functional block is defined by a *functional profile*. A functional profile is a template for functional block, and a functional block is an implementation of a functional profile. Functional profiles describe in detail the application-layer interface − including the network variables, configuration properties, defaults, and power-up behaviours − required on CNP devices for specific, commonly used control functions. This document defines standard functional profiles. Alternatively, manufacturers may define their own user functional profiles. User functional profiles shall be documented in manufacturer-specific resource files.

7.2 Functional Profile List

The following is a list of all Functional Profiles.

7.3 SFPTnodeObject (0)

Node Object: Allows the function of objects within a node to be monitored. Only one may exist on a node.

NV Member	Requirement	CNP Type	Direction	Description
1 "Object request" (nviRequest)	Mandatory	SNVT_obj_request	input	Requests a particular mode for a particular object in the device
2 "Object status" (nvoStatus)	Mandatory	SNVT_obj_status	output	Reports the status of requested object in the device
3 "Time setting" (nviTimeSet)	Optional	SNVT_time_stamp	input	Synchronize the node's internal real time clock with an external time source
4 "Alarm output" (nvoAlarm)	Optional	SNVT_alarm	output	Transmits alarm data for each object on a node whenever alarm occurs or is cleared, and upon request
5 "File request" (nviFileReq)	Optional	SNVT_file_req	input	Requests an operation on a particular file
6 "File status" (nvoFileStat)	Optional	SNVT_file_status	output	Reports the status of the last requested file operation
7 "File position" (nviFilePos)	Optional	SNVT_file_pos	input	Value used to control the position of the read/write pointer in a file
8 "Configuration parameter file directory address" (nvoFileDirectory)	Optional	SNVT_address	output	Address for file directory containing descriptors for configuration parameter files
9 "Alarm output 2" (nvoAlarm2)	Optional	SNVT_alarm_2	output	Transmits alarm data for each functional block on a device whenever an alarm occurs or is cleared, and upon request. Replaces nvoAlarm

NV Member	Requirement	CNP Type	Direction	Description
10 "Date event" (nviDateEvent)	Optional	SNVT_date_event	input	Reports the status of a schedule. Optional input for the node object for devices with Scheduler functional blocks
11 "Date resynchronization request" (nvoDateResync)	Optional	SNVT_switch	output	Requests an update for all defined exceptions via the nviDateEvent input. Required output from the node object for devices with Scheduler functional blocks
12 "Clear channel statistics" (nviClearStat)	Optional	SNVT_switch	input	Reset all channel statistic counters by setting nviClearStat to {100, ON} and back to {0, OFF}. Required on devices implementing the Channelmonitor
13 "Device uptime" (nvoUpTime)	Optional	SNVT_elapsed_tm	output	Time since the last reboot of the device
14 "Log file transfer request." (nviLogFxReq)	Optional	SNVT_log_fx_request	input	Requests a data log to be transferred via FTP. Shall be followed by a standard FTP request to get the data log file. Required on devices implementing the Data Logger functional profile that support data log transfer via FTP.
15 "Log status" (nvoLogStat)	Optional	SNVT_log_status	output	Reports the current status of a data log. Updated based on the cpLogNotificationThreshold value. Reports status only; alarms reported via Node Object nvoAlarm2 output. Required if the Node Object does not include an nvoLogStat output.
16 "Log file transfer status." (nvoLogFxStat)	Optional	SNVT_log_fx_status	output	Reports the status of a data log file transfer using FTP. Required on devices implementing the Data Logger functional profile that support data log transfer via FTP.
17 "Log status request." (nviLogReq)	Optional	SNVT_log_request	input	Requests the current status of a data log. Status is reported by the nvoLogStat output. Required on a device containing a Node Object with an nvoLogStat output if the device implements multiple Data Logger functional blocks.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTnwrkCnfg applying to SFPTnodeObject	Optional	none	not permitted	Indicates whether the node will configure itself, or expects a network manager
SCPTmaxSndT applying to nvoStatus	Optional	none	not permitted	Controls the maximum period of time before the object status is transmitted; Zero (0) means disabled
SCPTlocation applying to SFPTnodeObject	Optional	none	not permitted	Identifies the subsystem containing the device. The subsystem may be a simple location name or a hierarchical subsystem name. If a hierarchical subsystem name is specified, the subsystem hierarchy components shall be separated by periods ("."). Periods shall not otherwise be used. Other characters that cannot be used in a subsystem name are the backslash ("\"), colon (":"), forward slash ("/"), or double-quote characters. For very large networks, subsystem numbers may be used instead of subsystem names, for example: "1.2.29".
SCPTdevMajVer applying to SFPTnodeObject	Optional	constant	not permitted	The major version number for the device
SCPTdevMinVer applying to SFPTnodeObject	Optional	device specific, constant	not permitted	The minor version number for the device
SCPTgeoLocation applying to SFPTnodeObject	Optional	None	not permitted	GPS location where the physical device is located. This configuration property is mandatory if the SFPToutdoorLuminairController profile is implemented.

7.4 SFPTopenLoopSensor (1)

NV Member	Requirement	CNP Type	Direction	Description
1 "Value output" (nvoValue)	Mandatory	SNVT_xxx	output	Transmits the value from the sensor after conversion to correct units
2 "Preset input feedback" (nviPresetFb)	Optional	SNVT_preset	input	Receives preset function feedback information used to synchronize multiple source objects
3 "Raw hardware data output" (nvoRawHwData)	Optional	SNVT_count	output	Transmits the value obtained from the sensor prior to any transformation
4 "Preset output" (nvoPreset)	Optional	SNVT_preset	output	Used to program or control the preset of a destination object

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTlocation applying to SFPTopenLoopSensor	Optional	none	not permitted	Provides descriptive physical location information related to the object
SCPTgain applying to SFPTopenLoopSensor	Optional	none	not permitted	Used to calibrate external hardware via a multiplication factor
SCPToffset applying to SFPTopenLoopSensor	Optional	none	not permitted	Used to calibrate external hardware with additive offset after transformation
SCPTmaxRnge applying to nvoValue	Optional	none	not permitted	The maximum value limit of the output
SCPTminRnge applying to nvoValue	Optional	none	not permitted	The minimum value limit of the output

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTsndDelta applying to nvoValue	Optional	none	not permitted	The minimum change required to force transmission of the output value
SCPTmaxSndT applying to nvoValue	Optional	none	not permitted	The maximum period of time before a retransmission of value
SCPTminSndT applying to nvoValue	Optional	none	not permitted	The minimum period of time between retransmissions of value
SCPTinvrtOut applying to nvoValue	Optional	none	not permitted	Used to invert active polarity if the hardware sensor is a switch
SCPTtrnsTblX applying to SFPTopenLoopSensor	Optional	none	not permitted	Used in conjunction with Translation table Y to create a linearization table
SCPTtrnsTblY applying to SFPTopenLoopSensor	Optional	none	not permitted	Used in conjunction with Translation table X to create a linearization table
SCPThighLimit1 applying to nvoValue	Optional	none	not permitted	Used to remotely set the alarm high limit 1
SCPThighLimit2 applying to nvoValue	Optional	none	not permitted	Used to remotely set the alarm high limit 2
SCPTlowLimit1 applying to nvoValue	Optional	none	not permitted	Used to remotely set the alarm low limit 1
SCPTlowLimit2 applying to nvoValue	Optional	none	not permitted	Used to remotely set the alarm low limit 2
SCPTalrmSetT1 applying to nvoValue	Optional	none	not permitted	The time period that an alarm 1 condition shall exist before it is regarded as a valid alarm
SCPTalrmSetT2 applying to nvoValue	Optional	none	not permitted	The time period that an alarm 2 condition shall exist before it is regarded as a valid alarm

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTalrmClrT1 applying to nvoValue	Optional	none	not permitted	The time period that an alarm 1 condition shall not exist before it is regarded as a valid cleared alarm
SCPTalrmClrT2 applying to nvoValue	Optional	none	not permitted	The time period that an alarm 2 condition shall not exist before it is regarded as a valid cleared alarm
SCPThystHigh1 applying to nvoValue	Optional	none	not permitted	The hysteresis level for the value field of the alarm high 1 comparison threshold
SCPThystHigh2 applying to nvoValue	Optional	none	not permitted	The hysteresis level for the value field of the alarm high 2 comparison threshold
SCPThystLow1 applying to nvoValue	Optional	none	not permitted	The hysteresis level for the value field of the alarm low 1 comparison threshold
SCPThystLow2 applying to nvoValue	Optional	none	not permitted	The hysteresis level for the value field of the alarm low 2 comparison threshold
SCPTalrmIhbT applying to nvoValue	Optional	none	not permitted	Time period for which alarms are inhibited after enabling, node reset, or node is put on line
SCPTdefOutput applying to nvoValue	Optional	none	not permitted	The level the sensor should adopt in certain default conditions
SCPTovrBehave applying to SFPTopenLoopSensor	Optional	none	not permitted	The behaviour of a sensor when an override request is received
SCPTovrValue applying to nvoValue	Optional	none	not permitted	Sets the value a sensor should adopt when an object is overridden and behaviour is OV_SPECIFIED

7.5 SFPTclosedLoopSensor (2)

NV Member	Requirement	CNP Type	Direction	Description
1 "Value feedback input" (nviValueFb)	Mandatory	SNVT_xxx	input	Used to synchronize closed loop sensor objects when connected in multiple relationships
2 "Value output" (nvoValue)	Mandatory	SNVT_xxx	output	Transmits the value from the sensor after conversion to correct units
3 "Preset input feedback" (nviPresetFb)	Optional	SNVT_preset	input	Receives preset function feedback information used to synchronize multiple source objects
4 "Raw hardware data output" (nvoRawHwData)	Optional	SNVT_count	output	Transmits the value obtained from the sensor prior to any transformation
5 "Preset output" (nvoPreset)	Optional	SNVT_preset	output	Used to program or control the preset of a destination object

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTlocation applying to SFPTclosedLoopSensor	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTgain applying to SFPTclosedLoopSensor	Optional	none	not permitted	Used to calibrate external hardware via a multiplication factor
SCPToffset applying to SFPTclosedLoopSensor	Optional	none	not permitted	Used to calibrate external hardware with additive offset after transformation
SCPTmaxRnge applying to nvoValue	Optional	none	not permitted	The maximum value limit of the output

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTminRnge applying to nvoValue	Optional	none	not permitted	The minimum value limit of the output
SCPTsndDelta applying to nvoValue	Optional	none	not permitted	The minimum change required to force transmission of the output value
SCPTmaxSndT applying to nvoValue	Optional	none	not permitted	The maximum period of time before a retransmission of value
SCPTminSndT applying to nvoValue	Optional	none	not permitted	The minimum period of time between retransmissions of value
SCPTinvrtOut applying to nvoValue	Optional	none	not permitted	Used to invert active polarity if the hardware sensor is a switch
SCPTtrnsTblX applying to SFPTclosedLoopSensor	Optional	none	not permitted	Used in conjunction with Translation table Y to create a linearization table
SCPTtrnsTblY applying to SFPTclosedLoopSensor	Optional	none	not permitted	Used in conjunction with Translation table X to create a linearization table
SCPThighLimit1 applying to nvoValue	Optional	none	not permitted	Used to remotely set the alarm high limit 1
SCPThighLimit2 applying to nvoValue	Optional	none	not permitted	Used to remotely set the alarm high limit 2
SCPTlowLimit1 applying to nvoValue	Optional	none	not permitted	Used to remotely set the alarm low limit 1
SCPTlowLimit2 applying to nvoValue	Optional	none	not permitted	Used to remotely set the alarm low limit 2
SCPTalrmSetT1 applying to nvoValue	Optional	none	not permitted	The time period that an alarm 1 condition shall exist before it is regarded as a valid alarm

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTalrmSetT2 applying to nvoValue	Optional	none	not permitted	The time period that an alarm 2 condition shall exist before it is regarded as a valid alarm
SCPTalrmClrT1 applying to nvoValue	Optional	none	not permitted	The time period that an alarm 1 condition shall not exist before it is regarded as a valid cleared alarm
SCPTalrmClrT2 applying to nvoValue	Optional	none	not permitted	The time period that an alarm 2 condition shall not exist before it is regarded as a valid cleared alarm
SCPThystHigh1 applying to nvoValue	Optional	none	not permitted	The hysteresis level for the value field of the alarm high 1 comparison threshold
SCPThystHigh2 applying to nvoValue	Optional	none	not permitted	The hysteresis level for the value field of the alarm high 2 comparison threshold
SCPThystLow1 applying to nvoValue	Optional	none	not permitted	The hysteresis level for the value field of the alarm low 1 comparison threshold
SCPThystLow2 applying to nvoValue	Optional	none	not permitted	The hysteresis level for the value field of the alarm low 2 comparison threshold
SCPTalrmIhbT applying to nvoValue	Optional	none	not permitted	Time period for which alarms are inhibited after enabling, node reset, or node is put on line
SCPTdefOutput applying to nvoValue	Optional	none	not permitted	The level the sensor should adopt in certain default conditions
SCPTovrBehave applying to SFPTclosedLoopSensor	Optional	none	not permitted	The behaviour of a sensor when an override request is received
SCPTovrValue applying to nvoValue	Optional	none	not permitted	Sets the value a sensor should adopt when an object is overridden and behaviour is OV_SPECIFIED

7.6 SFPTopenLoopActuator (3)

NV Member	Requirement	CNP Type	Direction	Description
1 "Value input" (nviValue)	Mandatory	SNVT_xxx	input	Dictates the desired state of the actuator, determined by the specific application
2 "Preset input" (nviPreset)	Optional	SNVT_preset	input	Used to program or control the preset function
3 "Actual position feedback output" (nvoActPosnFb)	Optional	SNVT_xxx	output	Present position of the actuator, can be used as part of a control loop and for monitoring purposes
4 "Preset feedback output" (nvoPresetFb)	Optional	SNVT_preset	output	Transmits the setting associated with the current recalled or programmed preset

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTdefOutput applying to SFPTopenLoopActuator	Optional	none	not permitted	The level the actuator should adopt in certain default conditions
SCPTdriveT applying to SFPTopenLoopActuator	Optional	none	not permitted	Time to be taken by the actuator to move from one extreme to the other
SCPTlocation applying to SFPTopenLoopActuator	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTmaxRcvT applying to nviValue	Optional	none	not permitted	Maximum time that elapses after the last update to the input NV before actuator adopts the default output
SCPToffDely applying to SFPTopenLoopActuator	Optional	none	not permitted	Used for SNVT_switch or SNVT_lev_disc to determine the length of time the load remains energized after a change from ON to OFF is received

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTactFbDly applying to nvoActPosnFb	Optional	none	not permitted	The period for updating the feedback output when the actuator position does not match the requested position
SCPThighLimit1 applying to nviValue	Optional	none	not permitted	Used to remotely set the alarm high limit 1
SCPThighLimit2 applying to nviValue	Optional	none	not permitted	Used to remotely set the alarm high limit 2
SCPTlowLimit1 applying to nviValue	Optional	none	not permitted	Used to remotely set the alarm low limit 1
SCPTlowLimit2 applying to nviValue	Optional	none	not permitted	Used to remotely set the alarm low limit 2
SCPTalrmSetT1 applying to nviValue	Optional	none	not permitted	The time period that an alarm 2 condition shall exist before it is regarded as a valid alarm
SCPTalrmSetT2 applying to nviValue	Optional	none	not permitted	The time period that an alarm 2 condition shall exist before it is regarded as a valid alarm
SCPTalrmClrT1 applying to nviValue	Optional	none	not permitted	The time period that an alarm 1 condition shall not exist before it is regarded as a valid cleared alarm
SCPTalrmClrT2 applying to nviValue	Optional	none	not permitted	The time period that an alarm 2 condition shall not exist before it is regarded as a valid cleared alarm
SCPThystHigh1 applying to nviValue	Optional	none	not permitted	The hysteresis level for the value field of the alarm high 1 comparison threshold
SCPThystHigh2 applying to nviValue	Optional	none	not permitted	The hysteresis level for the value field of the alarm high 2 comparison threshold
SCPThystLow1 applying to nviValue	Optional	none	not permitted	The hysteresis level for the value field of the alarm low 1 comparison threshold
SCPThystLow2 applying to nviValue	Optional	none	not permitted	The hysteresis level for the value field of the alarm low 2 comparison threshold

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTalrmIhbT applying to nviValue	Optional	none	not permitted	Time period for which alarms are inhibited after enabling, node reset, or node is put on line
SCPTovrBehave applying to SFPTopenLoopActuator	Optional	none	not permitted	The behaviour of an actuator when an override request is received
SCPTovrValue applying to nviValue	Optional	none	not permitted	Sets the value an actuator should adopt when an object is overridden and behaviour is OV_SPECIFIED
SCPTtrnsTblX applying to SFPTopenLoopActuator	Optional	none	not permitted	Used in conjunction with Translation table Y to create a linearization table
SCPTtrnsTblY applying to SFPTopenLoopActuator	Optional	none	not permitted	Used in conjunction with Translation table X to create a linearization table

7.7 SFPTclosedLoopActuator (4)

NV Member	Requirement	CNP Type	Direction	Description
1 "Value input" (nviValue)	Mandatory	SNVT_xxx	input	Dictates the desired state of the actuator, determined by the specific application
2 "Value feedback output" (nvoValueFb)	Mandatory	SNVT_xxx	output	Transmits the current value of the input NV, used to synchronize source objects in multiple relationships
3 "Preset input" (nviPreset)	Optional	SNVT_preset	input	Used to program or control the preset function
4 "Actual position feedback output" (nvoActPosnFb)	Optional	SNVT_xxx	output	Present position of the actuator, can be used as part of a control loop and for monitoring purposes

NV Member	Requirement	CNP Type	Direction	Description
5 "Preset feedback output" (nvoPresetFb)	Optional	SNVT_preset	output	Transmits the setting associated with the current recalled or programmed preset

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTdefOutput applying to SFPTclosedLoopActuator	Optional	none	not permitted	The level the actuator should adopt in certain default conditions
SCPTdriveT applying to SFPTclosedLoopActuator	Optional	none	not permitted	Time to be taken by the actuator to move from one extreme to the other
SCPTlocation applying to SFPTclosedLoopActuator	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTmaxRcvT applying to nviValue	Optional	none	not permitted	Maximum time that elapses after the last update to the input NV before actuator adopts the default output
SCPToffDely applying to SFPTclosedLoopActuator	Optional	none	not permitted	Used for SNVT_switch or SNVT_lev_disc to determine the length of time the load remains energized after a change from ON to OFF is received
SCPTinFbDly applying to nvoValueFb	Optional	none	not permitted	The time period after the last update in a succession of changes to the input, before the feedback output is updated
SCPTactFbDly applying to nvoActPosnFb	Optional	none	not permitted	The period for updating the feedback output when the actuator position does not match the requested position
SCPThighLimit1 applying to nviValue	Optional	none	not permitted	Used to remotely set the alarm high limit 1

CNP Type	Requirement	Restrictions	Arrays	Description
SCPThighLimit2 applying to nviValue	Optional	none	not permitted	Used to remotely set the alarm high limit 2
SCPTlowLimit1 applying to nviValue	Optional	none	not permitted	Used to remotely set the alarm low limit 1
SCPTlowLimit2 applying to nviValue	Optional	none	not permitted	Used to remotely set the alarm low limit 2
SCPTalrmSetT1 applying to nviValue	Optional	none	not permitted	The time period that an alarm 1 condition shall exist before it is regarded as a valid alarm
SCPTalrmSetT2 applying to nviValue	Optional	none	not permitted	The time period that an alarm 2 condition shall exist before it is regarded as a valid alarm
SCPTalrmClrT1 applying to nviValue	Optional	none	not permitted	The time period that an alarm 1 condition shall not exist before it is regarded as a valid cleared alarm
SCPTalrmClrT2 applying to nviValue	Optional	none	not permitted	The time period that an alarm 2 condition shall not exist before it is regarded as a valid cleared alarm
SCPThystHigh1 applying to nviValue	Optional	none	not permitted	The hysteresis level for the value field of the alarm high 1 comparison threshold
SCPThystHigh2 applying to nviValue	Optional	none	not permitted	The hysteresis level for the value field of the alarm high 2 comparison threshold
SCPThystLow1 applying to nviValue	Optional	none	not permitted	The hysteresis level for the value field of the alarm low 1 comparison threshold
SCPThystLow2 applying to nviValue	Optional	none	not permitted	The hysteresis level for the value field of the alarm low 2 comparison threshold
SCPTalrmIhbT applying to nviValue	Optional	none	not permitted	Time period for which alarms are inhibited after enabling, node reset, or node is put on line

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTovrBehave applying to SFPTclosedLoopActuator	Optional	none	not permitted	The behaviour of an actuator when an override request is received
SCPTovrValue applying to nviValue	Optional	none	not permitted	Sets the value an actuator should adopt when an object is overridden and behaviour is OV_SPECIFIED
SCPTtrnsTblX applying to SFPTclosedLoopActuator	Optional	none	not permitted	Used in conjunction with Translation table Y to create a linearization table
SCPTtrnsTblY applying to SFPTclosedLoopActuator	Optional	none	not permitted	Used in conjunction with Translation table X to create a linearization table

7.8 SFPTcalendar (6)

NV Member	Requirement	CNP Type	Direction	Description
1 "Date event" (nvoDateEvent)	Optional	SNVT_date_event	output	Reports the status of a schedule
2 "Resynchronization request" (nviDateResynch)	Optional	SNVT_switch	input	Requests an update on the status of all defined schedules

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTeffectivePeriod applying to SFPTcalendar	Optional	modify when disabled	not permitted	Time period during which the calendar is effective
SCPTscheduleDates applying to SFPTcalendar	Mandatory	modify when disabled	required with dimension 2..65535	A range of dates with an optional qualifier that specifies when a schedule is active

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTobjMajVer applying to SFPTcalendar	Optional	constant	not permitted	The major version number for the object
SCPTobjMinVer applying to SFPTcalendar	Optional	device specific, constant	not permitted	The minor version number for the object
SCPTovrBehave applying to nvoDateEvent	Optional	none	not permitted	Behaviour of the nvoDateEvent output when an override request is received for the calendar
SCPTovrValue applying to nvoDateEvent	Optional	none	not permitted	The value the nvoDateEvent output should adopt when it is overridden and the value of Override Behavior is OV_SPECIFIED
SCPTscheduleName applying to SFPTcalendar	Optional	modify when disabled	required with dimension 2..255	Array of schedule names
SCPTminSendTime applying to nvoDateEvent	Optional	none	not permitted	Minimum period of time between automatic transmissions of the nvoDateEvent output. This throttles the series of outputs that occur whenever schedules are updated via nvoDateEvent

7.9 SFPTscheduler (7)

NV Member	Requirement	CNP Type	Direction	Description
1 "Present value" (nvoPresentValue)	Mandatory	SNVT_xxx	output	Indicates the current value of the schedule. The output is determined by the current time as well as the currently active schedules
2 "Enable" (nviEnable)	Optional	SNVT_switch	input	Enables the scheduler. The scheduler is enabled when the state value is one (1) and the level value is greater than zero (0)

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTdefOutput applying to nvoPresentValue	Mandatory	modify when disabled	not permitted	Specifies the nvoPresentValue output when no schedules are active, or when an override request is received for the functional block. The override behaviour may be defined by the optional Override Behaviour and Override Value configuration properties
SCPTdelayTime applying to nvoPresentValue	Optional	none	not permitted	Specifies a delay from the scheduled time. This allows multiple outputs within a Scheduler functional block, or multiple Scheduler functional blocks on a device, to share a common schedule but stagger on and off times to reduce peak load
SCPTeffectivePeriod applying to SFPTscheduler	Optional	modify when disabled	not permitted	Time period during which a functional block is effective.
SCPTmaxNVLength applying to nvoPresentValue	Optional	constant	not permitted	Maximum length of a type that may be assigned to the nvoPresentValue network variable
SCPTmaxSendTime applying to nvoPresentValue	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTnvType applying to nvoPresentValue	Optional	modify when disabled	not permitted	Assigns the network variable type for the nvoPresentValue network variable
SCPTobjMajVer applying to SFPTscheduler	Optional	constant	not permitted	The major version number for the object
SCPTobjMinVer applying to SFPTscheduler	Optional	device specific, constant	not permitted	The minor version number for the object
SCPTvalueName applying to nvoPresentValue	Optional	modify when disabled	required with dimension 2..65535	Used to create an array of value names for each of the values defined in a SCPTvalueDefinition array
SCPTovrBehave applying to nvoPresentValue	Optional	modify when disabled	not permitted	Defines the behaviour of the nvoPresentValue output when an override request is received

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTovrValue applying to nvoPresentValue	Optional	modify when disabled	not permitted	Sets the value the nvoPresentValue output should adopt when it is overridden and the value of Override Behaviour is OV_SPECIFIED
SCPTschedule applying to SFPTscheduler	Mandatory	modify when disabled	required with dimension 2..65535	Describes the attributes of a daily schedule definition
SCPTscheduleTimeValue applying to SFPTscheduler	Mandatory	modify when disabled	required with dimension 2..65535	Specifies the time and value for a scheduled event
SCPTweeklySchedule applying to SFPTscheduler	Optional	modify when disabled	not permitted	Identifies a schedule to be active for each day of the week
SCPTvalueDefinition applying to nvoPresentValue	Mandatory	modify when disabled	required with dimension 2..65535	Used to create an array of output values to be used for a schedule. A schedule time-value event specifies a value as an index into a SCPTvalueDefinition array
SCPTscheduleName applying to SFPTscheduler	Optional	modify when disabled	required with dimension 2..255	Used to create an array of schedule names

7.10 SFPTisiMonitorPoint (8)

NV Member	Requirement	CNP Type	Direction	Description
1 "Input value." (nviValue)	Mandatory	SNVT_xxx	input	Input value used in automatic connections initiated by the controller.

7.11 SFPTdataLogger (9)

NV Member	Requirement	CNP Type	Direction	Description
5 "Alarm status 2" (nvoLevAlarm)	Optional	SNVT_alarm_2	output	Transmits alarm data to a monitoring device. The alarm type is set to AL_NO_CONDITION if the level set by cpLogAlarmThreshold is not reached and to AL_ALM_CONDITION if the level has been reached or exceeded. The priority is set to PR_10.
2 "Enable" (nviEnable)	Optional	SNVT_switch	input	Enables data collection for the data log. Data collection is enabled when the state value is one (1). The level value is not used.
1 "Data value" (nviDataValue)	Optional	SNVT_xxx	input	Receives data to be collected in the data log. Optional because data sources may be internal to the device.
3 "Clear" (nviClear)	Optional	SNVT_switch	input	Clears the contents of the data log. The data log may also be cleared by a Node Object request.
4 "Log status" (nvoStatus)	Optional	SNVT_log_status	output	Reports the current status of a data log. Updated based on the cpLogNotificationThreshold value. Reports status only; alarms reported via Node Object nvoAlarm2 output. Required if the Node Object does not include an nvoLogStat output.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTlogCapacity applying to SFPTdataLogger	Mandatory	constant	not permitted	Specifies the total capacity of all data logs on a device. The size of each data log is specified by its cpLogSize value. The value is specified in bytes.
SCPTlogNotificationThreshold applying to SFPTdataLogger	Mandatory	none	not permitted	Specifies the percentage change in log level required to trigger an update to the Data Log Status (nvoStatus) output.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTlogSize applying to SFPTdataLogger	Mandatory	modify when disabled	not permitted	Capacity of a data log.
SCPTlogType applying to SFPTdataLogger	Mandatory	modify when disabled	not permitted	Specifies the method used to store data in a data log.
SCPTfanInEnable applying to nvoLevAlarm	Optional	modify when disabled	not permitted	Enables fan-in of multiple data sources. When True, the application examines the source address of each input value and uses it to determine the data source of the update.
SCPTobjMajVer applying to SFPTdataLogger	Optional	constant	not permitted	Major version number for the data logger implementation. This value is incremented when the network interface for the functional block changes.
SCPTobjMinVer applying to SFPTdataLogger	Optional	device specific, constant	not permitted	Minor version number for the data logger implementation. This value is incremented when the functional block implementation changes without changing the functional block network interface. The minor version is reset to zero and the cpObjMajVer value is incremented when the functional block network interface changes.
SCPTlogTimestampEnable applying to SFPTdataLogger	Optional	modify when disabled	not permitted	Enables time stamping of each data value. When True, the data logger includes a timestamp of the receipt time for each value received by the data logger.
SCPTlogHighLimit applying to nvoLevAlarm	Optional	none	not permitted	Enables logging of data greater or equal to the specified value. All other data is ignored, with the exception that data that is less than or equal to a valid SCPTlogLowLimit value is also logged. All data is logged if the SCPTlogEnableHighLimit and SCPTlogEnableLowLimit values are both invalid.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTlogLowLimit applying to nvoLevAlarm	Optional	none	not permitted	Enables logging of data less than or equal to the specified value. All other data is ignored, with the exception that data that is greater than or equal to a valid SCPTlogHighLimit value is also logged. All data is logged if the SCPTlogEnableHighLimit and SCPTlogEnableLowLimit values are both invalid.
SCPTmaxFanIn applying to nvoLevAlarm	Optional	constant	not permitted	Specifies the maximum number of data sources that may be connected to a network variable. The functional block determines data sources by examining the source address of each update.
SCPTlogMinDeltaTime applying to nvoLevAlarm	Optional	modify when disabled	not permitted	Minimum amount of time between logged values. This is used to throttle data entry into a data log. When a data value is logged, a subsequent update to the data value is not logged until the time specified by this value has elapsed. If additional updates are received during this time, the older values are discarded and are not stored in the data log. Time of receipt is ignored if the value of this configuration property is zero or invalid.
SCPTlogMinDeltaValue applying to nvoLevAlarm	Optional	modify when disabled	not permitted	Minimum amount of time between logged values. This is used to throttle data entry into a data log. When a data value is logged, a subsequent update to the data value is not logged until the time specified by this value has elapsed. If additional updates are received during this time, the older values are discarded and are not stored in the data log. Time of receipt is ignored if the value of this configuration property is zero or invalid.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to nviClear	Optional	none	not permitted	Maximum period of time that expires before the Data Logger functional block automatically transmits the current value of the nvoStatus output network variable. This provides a heartbeat output that can be used by destination functional blocks to ensure that the functional block is still healthy.
SCPTnvType applying to nvoLevAlarm	Optional	modify when disabled	not permitted	Network variable type for network variables that support changeable types
SCPTpollRate applying to nvoLevAlarm	Optional	modify when disabled	not permitted	Specifies the poll rate for each data source. When this value is greater than zero, the functional block polls each of the data sources identified in the source address array at the rate specified by this value.
SCPTsourceAddress applying to nvoLevAlarm	Optional	device specific, constant, modify when disabled	required with dimension 2..32385	Specifies a source address or element of an array of source addresses for and input to a functional block.
SCPTmaxSendTime applying to nvoStatus	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxNVLength applying to nvoLevAlarm	Optional	modify when disabled	not permitted	Maximum length of a type that may be assigned to the nviDataValue input. Also requires implementation of the cpNvType configuration property.
SCPTlogAlarmThreshold applying to SFPTdataLogger	Optional	none	not permitted	Specifies the log level required to trigger an alarm condition for the data logger.

7.12 SFPTschedulerSimple (17)

NV Member	Requirement	CNP Type	Direction	Description
5 "Alarm status 2" (nvoLevAlarm)	Optional	SNVT_alarm_2	output	Transmits alarm data to a monitoring device. The alarm type is set to AL_NO_CONDITION if the level set by cpLogAlarmThreshold is not reached and to AL_ALM_CONDITION if the level has been reached or exceeded. The priority is

				set to PR_10.
2 "Enable" (nviEnable)	Optional	SNVT_switch	input	Enables data collection for the data log. Data collection is enabled when the state value is one (1). The level value is not used.
1 "Data value" (nviDataValue)	Optional	SNVT_xxx	input	Receives data to be collected in the data log. Optional because data sources may be internal to the device.
3 "Clear" (nviClear)	Optional	SNVT_switch	input	Clears the contents of the data log. The data log may also be cleared by a Node Object request.
Optional	4 "Log status" (nvoStatus)	SNVT_log_status	output	Reports the current status of a data log. Updated based on the cpLogNotificationThreshold value. Reports status only; alarms reported via Node Object nvoAlarm2 output. Required if the Node Object does not include an nvoLogStat output.

7.13 SFPTchannelMonitor (132)

NV Member	Requirement	CNP Type	Direction	Description
2 "Bandwidth utilization" (nvoIvalBandUtl)	Mandatory	SNVT_lev_cont	output	Bandwidth utilization percentage during last interval.
3 "CRC error rate" (nvoIvalCrcErr)	Mandatory	SNVT_lev_cont	output	Percentage of packets with CRC errors during last interval.
4 "Missed packet rate" (nvoIvalMissed)	Mandatory	SNVT_lev_cont	output	Percentage of missed packets during last interval.
5 "Packet rate" (nvoIvalPkts)	Mandatory	SNVT_count_32	output	Number of packets per second during last interval.
6 "Total CRC errors" (nvoTotalCrcErr)	Mandatory	SNVT_count_32	output	Total number of CRC errors since power-up, reset, or statistics cleared.

NV Member	Requirement	CNP Type	Direction	Description
7 "Total missed packets" (nvoTotalMissed)	Mandatory	SNVT_count_32	output	Total number of missed packets since power-up, reset, or statistics cleared.
8 "Total packets" (nvoTotalPkts)	Mandatory	SNVT_count_32	output	Total number of packets since power-up, reset, or statistics cleared.
1 "Elapsed Time" (nvoElapsedTime)	Mandatory	SNVT_elapsed_tm	output	Time since the device was rebooted or since the statistics for this interface where reset.
11 "Maximum CRC error rate" (nvoMaxCrcErr)	Optional	SNVT_lev_cont	output	Maximum percentage of CRC errors in any interval, as reported by nvoIvalCrcErr, since power-up, reset, or statistics cleared.
10 "Maximum bandwidth utilization" (nvoMaxBandUtl)	Optional	SNVT_lev_cont	output	Maximum percentage of bandwidth utilization in any interval, as reported by nvoIvalBandUtl, since power-up, reset, or statistics cleared.
12 "Maximum missed packet rate" (nvoMaxMissed)	Optional	SNVT_lev_cont	output	Maximum percentage of missed packets in any interval, as reported by nvoIvalMissed, since power-up, reset, or statistics cleared.
13 "Maximum packet rate" (nvoMaxPkts)	Optional	SNVT_count_32	output	Maximum number of packets per second in any interval, as reported by nvoIvalPackets, since power-up, reset, or statistics cleared.
14 "Missed preamble rate" (nvoIvalMisPre)	Optional	SNVT_count_32	output	Number of missed preambles per second during the last interval.
15 "Total missed preambles" (nvoTotalMisPre)	Optional	SNVT_count_32	output	Total number of missed preambles since power-up, reset, or statistic cleared.
16 "Maximum missed preamble rate" (nvoMaxMisPre)	Optional	SNVT_count_32	output	Maximum number of missed preambles per second in any interval, as reported by nvoIvalMisPre, since power-up, reset, or statistics cleared.

NV Member	Requirement	CNP Type	Direction	Description
17 "Channel alarm" (nvoChnlAlarm)	Optional	SNVT_switch	output	Indicates if the associated channel is in alarm state or shows a high error rate.
18 "Channel alarm ratio" (nvoChnlAlmRat)	Optional	SNVT_lev_cont	output	Percentage of intervals the associated channel was in alarm state or had a high error rate since power-up, reset, or statistics cleared.
19 "Link power supply status" (nvoLinkPower)	Optional	SNVT_switch	output	Indicates whether the Link Power Supply voltage is in the specified range.
9 "Average packets" (nvoAvgPkts)	Mandatory	SNVT_count_32	output	Average packets per second since power-up, reset, or statistics cleared. Rollover is implementation dependent.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTobjMajVer applying to SFPTchannelMonitor	Optional	constant	not permitted	The major version number for the object
SCPTobjMinVer applying to SFPTchannelMonitor	Optional	constant	not permitted	The minor version number for the object
SCPTifaceDesc applying to SFPTchannelMonitor	Optional	constant	not permitted	Human readable description of the interface the functional block is assigned to.
SCPTmonInterval applying to SFPTchannelMonitor	Optional	none	not permitted	This configuration property defines the interval over which statistics are collected and averages are calculated.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPThighLimit1 applying to nvoIvalCrcErr	Optional	none	not permitted	If this limit is implemented and is enabled via nciBandUtlLimHighEnable, if the alarm type was not AL_HIGH_LMT_ALM_1 and the value for the last interval was greater than nciBandUtlLimHigh, then the alarm type becomes AL_HIGH_LMT_ALM_1 and nvoChnlAlarm will be set to True.If the alarm type was AL_HIGH_LMT_ALM_1 and the value drops below nciBandUtlLimHigh minus nciBandUtlHystHigh, then the alarm type becomes AL_HIGH_LMT_CLR_1 and nvoChnlAlarm will be set to False. The hysteresis value is used only to subtract from the limit value.
SCPThighLimit1Enable applying to nvoIvalCrcErr	Optional	none	not permitted	Determines whether nciBandUtlLimHigh is enabled.
SCPThystHigh1 applying to nvoIvalCrcErr	Optional	none	not permitted	Hysteresis value for the high alarm nciBandUtlLimHigh.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTlowLimit1 applying to nvoIvalCrcErr	Optional	none	not permitted	The low alarm limit can be useful to detect that, for instance, a certain level of background heartbeat traffic expected on a channel has slowed or stopped-which indicates a power, channel wiring, or other problem. If the low alarm limit is enabled with nciBandUtlLimLowEnable, also set the nciBandUtlLimLow property to a non-zero value that is slightly below the expected level of background heartbeat traffic. If the alarm type was not AL_LOW_LMT_ALM_1 and the value is now less than nciBandUtlLimLow, then the alarm type becomes AL_LOW_LMT_ALM_1 and nvoChnlAlarm will be set to True.If the alarm type was AL_LOW_LMT_ALM_1 and the value rises above nciBandUtlLimLow plus nciBandUtlHystLow, then the alarm type becomes AL_LOW_LMT_CLR_1 and nvoChnlAlarm will be set to False. The hysteresis value is used only to add to the limit value.
SCPTlowLimit1Enable applying to nvoIvalCrcErr	Optional	none	not permitted	Determines whether nciBandUtlLimLow is enabled.
SCPThystLow2 applying to nvoIvalCrcErr	Optional	none	not permitted	Hysteresis value for the low alarm nciBandUtlLimLow.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPThighLimit1 applying to nvoIvalMissed	Optional	none	not permitted	If this limit is implemented and is enabled via nciCrcErrLimHighEnable, if the alarm type was not AL_HIGH_LMT_ALM_1 and the value for the last interval was greater than nciCrcErrLimHigh, then the alarm type becomes AL_HIGH_LMT_ALM_1 and nvoChnlAlarm will be set to True.If the alarm type was AL_HIGH_LMT_ALM_1 and the value drops below nciCrcErrLimHigh minus nciCrcErrHystHigh, then the alarm type becomes AL_HIGH_LMT_CLR_1 and nvoChnlAlarm will be set to False. The hysteresis value is used only to subtract from the limit value.
SCPThighLimit1Enable applying to nvoIvalMissed	Optional	none	not permitted	Determines whether nciCrcErrLimHigh is enabled.
SCPThystHigh1 applying to nvoIvalMissed	Optional	none	not permitted	Hysteresis value for the high alarm nciCrcErrLimHigh.
SCPThighLimit1 applying to nvoIvalPkts	Optional	none	not permitted	If this limit is implemented and is enabled via nciMissedLimHighEnable, if the alarm type was not AL_HIGH_LMT_ALM_1 and the value for the last interval was greater than nciMissedLimHigh, then the alarm type becomes AL_HIGH_LMT_ALM_1 and nvoChnlAlarm will be set to True.If the alarm type was AL_HIGH_LMT_ALM_1 and the value drops below nciMissedLimHigh minus nciMissedHystHigh, then the alarm type becomes AL_HIGH_LMT_CLR_1 and nvoChnlAlarm will be set to False. The hysteresis value is used only to subtract from the limit value.
SCPThighLimit1Enable applying to nvoIvalPkts	Optional	none	not permitted	Determines whether nciMissedLimHigh is enabled.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPThystHigh1 applying to nvoIvalPkts	Optional	none	not permitted	Hysteresis value for the high alarm nciMissedLimHigh.
SCPThighLimit1 applying to nvoTotalMisPre	Optional	none	not permitted	If this limit is implemented and is enabled via nciMisPreLimHighEnable, if the alarm type was not AL_HIGH_LMT_ALM_1 and the value for the last interval was greater than nciMisPreLimHigh, then the alarm type becomes AL_HIGH_LMT_ALM_1 and nvoChnlAlarm will be set to True.If the alarm type was AL_HIGH_LMT_ALM_1 and the value drops below nciMisPreLimHigh minus nciMisPreHystHigh, then the alarm type becomes AL_HIGH_LMT_CLR_1 and nvoChnlAlarm will be set to False. The hysteresis value is used only to subtract from the limit value.
SCPThighLimit1Enable applying to nvoTotalMisPre	Optional	none	not permitted	Determines whether nciMisPreLimHigh is enabled.
SCPThystHigh1 applying to nvoTotalMisPre	Optional	none	not permitted	Hysteresis value for the high alarm nciMisPreLimHigh.
SCPTlinkPowerDetectEnable applying to nvoAvgPkts	Optional	none	not permitted	Determines, whether link power detection is enabled. If yes, nvoLinkPower indicates existence of link power voltage.

7.14 SFPTdeviceMonitor (136)

NV Member	Requirement	CNP Type	Direction	Description
1 (nvoDeviceAlarm)	Mandatory	SNVT_state_64	output	
3 "Scan Time" (nvoRingALastDev)	Optional	SNVT_count	output	Device Alarm

NV Member	Requirement	CNP Type	Direction	Description
4 "Scan Time" (nvoRingBLastDev)	Optional	SNVT_count	output	States of all monitored device
5 "Devices responding on side A" (nvoRingARcv)	Optional	SNVT_state_64	output	Shows which devices can be reached via side A.
7 "Devices responding on side B" (nvoRingBRcv)	Optional	SNVT_state_64	output	Shows which devices can be reached via side B

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to nvoDeviceAlarm	Mandatory	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTscanTime applying to SFPTdeviceMonitor	Mandatory	none	not permitted	Duration in which all devices are being queried.
SCPTobjMajVer applying to SFPTdeviceMonitor	Optional	constant	not permitted	The major version number for the object
SCPTobjMinVer applying to SFPTdeviceMonitor	Optional	constant	not permitted	The minor version number for the object
SCPTdevListEntry applying to SFPTdeviceMonitor	Optional	modify when disabled	required with dimension 128	Device list entry containing the address of the device to be monitored.
SCPTdevListDesc applying to SFPTdeviceMonitor	Optional	modify when disabled	required with dimension 128	Human readable description for an entry in the device list.

7.15 SFPTchannelContinuityMonitor (137)

NV Member	Requirement	CNP Type	Direction	Description
1 "Channel State OK" (nvoChannelOK)	Mandatory	SNVT_switch	output	Signals if the channel is OK or interrupted.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to nvoChannelOK	Mandatory	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTminSendTime applying to nvoChannelOK	Optional	none	not permitted	The minimum period of time between consecutive transmissions of the current value
SCPTobjMajVer applying to SFPTchannelContinuityMonitor	Optional	constant	not permitted	The major version number for the object
SCPTobjMinVer applying to SFPTchannelContinuityMonitor	Optional	constant	not permitted	The minor version number for the object

7.16 SFPTstaticProgrammable (410)

NV Member	Requirement	CNP Type	Direction	Description
1 "Program Status" (nvoProgramStatus)	Mandatory	SNVT_program_status	output	This input network variable is used to indicate the current status of the Program object.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTprogSelect applying to SFPTstaticProgrammable	Optional	none	not permitted	This configuration property specifies which program is selected to be loaded, if the device can store multiple programs.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTobjMajVer applying to SFPTstaticProgrammable	Optional	constant	not permitted	The major version number for the object
SCPTobjMinVer applying to SFPTstaticProgrammable	Optional	device specific, constant	not permitted	The minor version number for the object
SCPTprogSourceLocation applying to SFPTstaticProgrammable	Optional	device specific, constant	not permitted	This configuration property specifies the location from where the source file was downloaded to the device.
SCPTprogFileIndexes applying to SFPTstaticProgrammable	Optional	constant	not permitted	This configuration property may be implemented if the program data is stored in one or more files on the device.
SCPTprogCmdHistory applying to SFPTstaticProgrammable	Optional	device specific, constant	permitted with dimension 0..9999	This configuration property contains a read-only list of the most recent commands to the device, along with a time stamp for each.
SCPTprogStateHistory applying to SFPTstaticProgrammable	Optional	device specific, constant	permitted with dimension 0..9999	This configuration property contains a read-only list of the most recent state of the device, along with a time stamp for each.
SCPTnsdsFbIndex applying to SFPTstaticProgrammable	Optional	none	not permitted	The Functional Block index within the Node Self-Documentation string to which this program applies
SCPTprogErrorHistory applying to SFPTstaticProgrammable	Optional	device specific, constant	permitted with dimension 0..9999	Log of recent errors, with time stamp
SCPTprogName applying to SFPTstaticProgrammable	Mandatory	device specific, constant	not permitted	This configuration property specifies the human-readable name of the currently loaded program.
SCPTprogRevision applying to SFPTstaticProgrammable	Mandatory	device specific, constant	not permitted	This configuration property specifies the revision number of the currently loaded program.
SCPTnvUsage applying to SFPTstaticProgrammable	Optional	none	not permitted	The SCPTnvUsage CPs shall be used to indicate whether the NVs are in use by the loaded program

7.17 SFPTanalogInput (520)

NV Member	Requirement	CNP Type	Direction	Description
1 "Analog value" (nvoAnalog)	Mandatory	SNVT_lev_percent	output	The value of the analogue input signal converted to a level-percentage

7.18 SFPTanalogOutput (521)

NV Member	Requirement	CNP Type	Direction	Description
1 "Analog value" (nviAnalog)	Mandatory	SNVT_lev_percent	input	The value of the analogue output signal expressed as a level-percentage

7.19 SFPTlightSensor (1010)

NV Member	Requirement	CNP Type	Direction	Description
1 "Light level output" (nvoLuxLevel)	Mandatory	SNVT_lux	output	Monitors the value of the hardware input

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTlocation applying to SFPTlightSensor	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTreflection applying to SFPTlightSensor	Optional	none	not permitted	Adjusts the internal gain factor for the measured illumination level
SCPTfieldCalib applying to SFPTlightSensor	Optional	none	not permitted	Used by the light sensor to self-calibrate the hardware

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTminSendTime applying to nvoLuxLevel	Optional	none	not permitted	Minimum period between output NV transmissions (maximum transmission rate)
SCPTmaxSendTime applying to nvoLuxLevel	Optional	none	not permitted	Maximum period of time that expires before the object automatically transmits the present value of the lux level output NV
SCPTminDeltaLevel applying to nvoLuxLevel	Optional	none	not permitted	Used to control the % amount by which the value shall change before lux level output NV is transmitted

7.20 SFPTglobalSolarRadiation (1015)

NV Member	Requirement	CNP Type	Direction	Description
1 "Watts per square meter" (nvoGlobalRad)	Mandatory	SNVT_Wm2_p	output	This output network variable reports the Solar Global Radiation with W/m^2

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to nvoGlobalRad	Mandatory	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTminSendTime applying to nvoGlobalRad	Mandatory	none	not permitted	The minimum period of time between consecutive transmissions of the current value
SCPTsndDelta applying to nvoGlobalRad	Mandatory	none	not permitted	The minimum change required to force transmission of the output value
SCPToffset applying to nvoGlobalRad	Optional	none	not permitted	Used to calibrate the level that the associated output network variable should adopt after any translation table or gain factor.
SCPTgain applying to nvoGlobalRad	Optional	none	not permitted	This parameter is used to calibrate the external hardware.

7.21 SFPTpressureSensor (1030)

NV Member	Requirement	CNP Type	Direction	Description
1 "Standard Pressure Output" (nvoPress)	Mandatory	SNVT_press	output	This output variable reports the pressure of the sensor using the Kilo Pascal pressure SNVT_press.
2 "High Precision Pressure Output" (nvoPrecisePress)	Optional	SNVT_press_p	output	This output variable reports the pressure of the sensor using the high precision pressure SNVT_press_p.
3 "Floating Point Pressure Output" (nvoFloatPress)	Optional	SNVT_press_f	output	This output variable reports the pressure of the sensor using the Floating point pressure SNVT.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to SFPTpressureSensor	Mandatory	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTminSendTime applying to SFPTpressureSensor	Mandatory	none	not permitted	The minimum period of time between consecutive transmissions of the current value
SCPTsndDelta applying to SFPTpressureSensor	Mandatory	none	not permitted	The minimum change required to force transmission of the output value
SCPToffset applying to nvoPress	Optional	none	not permitted	Used to calibrate the level that the associated output network variable should adopt after any translation table or gain factor.
SCPToffset applying to nvoPrecisePress	Optional	none	not permitted	Used to calibrate the level that the associated output network variable should adopt after any translation table or gain factor.
SCPToffset applying to nvoFloatPress	Optional	none	not permitted	Used to calibrate the level that the associated output network variable should adopt after any translation table or gain factor.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTlocation applying to SFPTpressureSensor	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTmaxSendTime applying to nvoPress	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoPrecisePress	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoFloatPress	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTminSendTime applying to nvoPress	Optional	none	not permitted	The minimum period of time between consecutive transmissions of the current value
SCPTminSendTime applying to nvoPrecisePress	Optional	none	not permitted	The minimum period of time between consecutive transmissions of the current value
SCPTminSendTime applying to nvoFloatPress	Optional	none	not permitted	The minimum period of time between consecutive transmissions of the current value
SCPTsndDelta applying to nvoPress	Optional	none	not permitted	The minimum change required to force transmission of the output value
SCPTsndDelta applying to nvoPrecisePress	Optional	none	not permitted	The minimum change required to force transmission of the output value
SCPTsndDelta applying to nvoFloatPress	Optional	none	not permitted	The minimum change required to force transmission of the output value

7.22 SFPThvacTempSensor (1040)

NV Member	Requirement	CNP Type	Direction	Description
1 "Temperature" (nvoHVACTemp)	Mandatory	SNVT_temp_p	output	The temperature of the sensor using HVAC temperature data type
2 "Temperature" (nvoFixPtTemp)	Optional	SNVT_temp	output	The temperature of the sensor using fixed point data type
3 "Temperature" (nvoFloatTemp)	Optional	SNVT_temp_f	output	The temperature of the sensor using floating point data type

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to SFPThvacTempSensor	Mandatory	none	not permitted	Controls the maximum period of time before the output values are transmitted
SCPTminSendTime applying to SFPThvacTempSensor	Mandatory	none	not permitted	Controls the minimum period of time between output value transmissions
SCPTminDeltaTemp applying to nvoHVACTemp	Mandatory	none	not permitted	The minimum change required to force transmission of the output value
SCPToffsetTemp applying to nvoHVACTemp	Optional	none	not permitted	Used to calibrate external hardware with additive offset after transformation

7.23 SFPTfrostSensor (1042)

NV Member	Requirement	CNP Type	Direction	Description
1 "Frost Data" (nvoFrost)	Mandatory	SNVT_switch	output	This output network variable provides the state of the frost detector.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to nvoFrost	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTminSendTime applying to nvoFrost	Optional	none	not permitted	The minimum period of time between consecutive transmissions of the current value

7.24 SFPThvacRelativeHumiditySensor (1050)

NV Member	Requirement	CNP Type	Direction	Description
1 "Relative humidity" (nvoHVACRH)	Mandatory	SNVT_lev_percent	output	The relative humidity using the level-percent data type
2 "Relative humidity" (nvo8bitRH)	Optional	SNVT_lev_cont	output	The relative humidity using the short fixed point data type
3 "Relative humidity" (nvoFloatRH)	Optional	SNVT_lev_cont_f	output	The relative humidity using floating point data type

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to SFPThvacRelativeHumiditySensor	Mandatory	none	not permitted	Controls the maximum period of time before the output values are transmitted
SCPTminSendTime applying to SFPThvacRelativeHumiditySensor	Mandatory	none	not permitted	Controls the minimum period of time between output value transmissions
SCPTminDeltaRH applying to nvoHVACRH	Mandatory	none	not permitted	The minimum change required to force transmission of the output value
SCPToffsetRH applying to nvoHVACRH	Optional	none	not permitted	Used to calibrate external hardware with additive offset after transformation

7.25 SFPTrainSensor (1051)

NV Member	Requirement	CNP Type	Direction	Description
1 "Rain Data" (nvoRain)	Mandatory	SNVT_switch	output	This output network variable provides the state of the rain detector.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTminSendTime applying to nvoRain	Optional	none	not permitted	The minimum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoRain	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value

7.26 SFPToccupancySensor (1060)

NV Member	Requirement	CNP Type	Direction	Description
1 "Occupancy" (nvoOccup)	Mandatory	SNVT_occupancy	output	Provides the qualified state of the hardware sensor output

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTlocation applying to SFPToccupancySensor	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTmaxSendTime applying to SFPToccupancySensor	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTdebounce applying to SFPToccupancySensor	Optional	none	not permitted	The debouncing time to generate the detection envelope

7.27 SFPTisiOccupancySensor (1061)

NV Member	Requirement	CNP Type	Direction	Description
1 "Occupancy state" (nvoSwitch)	Mandatory	SNVT_switch_2	output	Reports the occupancy state for a zone. Occupancy is reported when detected. Feedback from other connected occupancy sensors within the zone is monitored -- the unoccupied state is only propagated if no connected occupancy sensors are reporting occupancy.
2 "Occupancy feedback input" (nviSwitchFb)	Mandatory	SNVT_switch_2	input	Used to synchronise multiple occupancy sensors within a zone
3 "Occupancy" (nvoOccup)	Mandatory	SNVT_occupancy	output	Provides the qualified state of the hardware sensor output
5 "Test control" (nviTest)	Mandatory	SNVT_switch_2	input	Controls test mode. Set to SW_SET_ON to enable test mode--turning off the hold time and ignoring the feedback input; set to SW_SET_OCCUPIED to force the output to occupied; set to SW_SET_UNOCCUPIED to force output to unoccupied; set to SW_SET_OFF to enable normal operation. Default is SW_SET_OFF.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTlocation applying to SFPTisiOccupancy Sensor	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTname1 applying to SFPTisiOccupancy Sensor	Mandatory	none	not permitted	Part 1 of the name of the functional block to be used by optional user interface applications. May optionally used with SCPTname2 and SCPTname3. Shall be implemented as a configuration network variable.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTname2 applying to SFPTisiOccupancySensor	Mandatory	none	not permitted	Part 2 of the name of the functional block to be used by optional user interface applications. Shall be used with SCPTname1 and may optionally be used with SCPTname3. This part is concatenated after part 1, and may optionally be followed by part 3. Shall be implemented as a configuration network variable.
SCPTname3 applying to SFPTisiOccupancySensor	Optional	none	not permitted	Part 3 of the name of the functional block to be used by optional user interface applications. Shall be used with SCPTname1 and SCPTname2. This part, if present, is concatenated with parts 1 and 2. Shall be implemented as a configuration network variable.
SCPTobjMajVer applying to SFPTisiOccupancySensor	Optional	none	not permitted	Major version number for the associated functional block. See also cpFbMinVer.
SCPTobjMinVer applying to SFPTisiOccupancySensor	Optional	none	not permitted	Minor version number for the associated functional block. See also cpFbMajVer.
SCPTdebounce applying to nvoSwitch	Mandatory	none	not permitted	The interval after a change to the occupied state that the occupancy sensor input is ignored. Shall be implemented as a configuration network variable.
SCPTholdTime applying to nvoSwitch	Mandatory	none	not permitted	Hold time for the nvoSwitch occupied state after there is no occupancy detected. The hold time timer is retriggered each time the sensor reports the area as occupied. Shall be implemented as a configuration network variable.
SCPTmaxSendTime applying to SFPTisiOccupancySensor	Mandatory	none	not permitted	The maximum period of time between consecutive transmissions of the current value. If this value is set to the invalid value, the heartbeat interval will be 2 min when occupied and 20 min when unoccupied -- this is the default behaviour. Shall be implemented as a configuration network variable.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTinFbDly applying to nvoSwitch	Mandatory	none	not permitted	The time period after a change in the calculated state from occupied to unoccupied before the unoccupied state is reported. Shall be implemented as a configuration network variable.

7.28 SFPTco2Sensor (1070)

NV Member	Requirement	CNP Type	Direction	Description
1 "CO2 level" (nvoCO2ppm)	Mandatory	SNVT_ppm	output	The CO2 level in parts-per-million
2 "Temperature" (nvofloatCO2)	Optional	SNVT_ppm_f	output	The CO2 level in parts-per-million using floating point data type

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to SFPTco2Sensor	Mandatory	none	not permitted	Controls the maximum period of time before the output values are transmitted
SCPTminSendTime applying to SFPTco2Sensor	Mandatory	none	not permitted	Controls the minimum period of time between output value transmissions
SCPTminDeltaCO2 applying to nvoCO2ppm	Mandatory	none	not permitted	The minimum change required to force transmission of the output value
SCPToffsetCO2 applying to nvoCO2ppm	Optional	none	not permitted	Used to calibrate external hardware with additive offset after transformation

7.29 SFPTairVelocitySensor (1083)

NV Member	Requirement	CNP Type	Direction	Description
1 "Air Velocity" (nvoAirVelocity)	Mandatory	SNVT_speed_mil	output	This output network variable reports the velocity detected by the sensor.
2 "Duct Flow Rate" (nvoAirFlow)	Optional	SNVT_flow	output	This output network variable provides the flow volume flowing through the duct.
3 "Flow Rate" (nvoAirFlowFloat)	Optional	SNVT_flow_f	output	This output network variable provides the flow volume flowing through the duct.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to SFPTairVelocitySensor	Mandatory	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTminSendTime applying to SFPTairVelocitySensor	Mandatory	none	not permitted	The minimum period of time between consecutive transmissions of the current value
SCPTsndDelta applying to SFPTairVelocitySensor	Mandatory	none	not permitted	The minimum change required to force transmission of the output value
SCPToffset applying to SFPTairVelocitySensor	Optional	none	not permitted	Used to calibrate the level that the associated output network variable should adopt after any translation table or gain factor.
SCPTductArea applying to nvoAirVelocity	Optional	none	not permitted	The duct area used to calculate the air flow, relevant only for VAV actuators / controllers
SCPTductArea applying to nvoAirFlow	Optional	none	not permitted	The duct area used to calculate the air flow, relevant only for VAV actuators / controllers

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTgain applying to SFPTairVelocitySensor	Optional	none	not permitted	This parameter is used to calibrate the external hardware.
SCPTmaxSendTime applying to nvoAirVelocity	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoAirFlow	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoAirFlowFloat	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTminSendTime applying to nvoAirVelocity	Optional	none	not permitted	The minimum period of time between consecutive transmissions of the current value
SCPTminSendTime applying to nvoAirFlow	Optional	none	not permitted	The minimum period of time between consecutive transmissions of the current value
SCPTminSendTime applying to nvoAirFlowFloat	Optional	none	not permitted	The minimum period of time between consecutive transmissions of the current value
SCPTsndDelta applying to nvoAirVelocity	Optional	none	not permitted	The minimum change required to force transmission of the output value
SCPTsndDelta applying to nvoAirFlow	Optional	none	not permitted	The minimum change required to force transmission of the output value
SCPTsndDelta applying to nvoAirFlowFloat	Optional	none	not permitted	The minimum change required to force transmission of the output value
SCPTductArea applying to SFPTairVelocitySensor	Optional	none	not permitted	The duct area used to calculate the air flow, relevant only for VAV actuators / controllers

7.30 SFPTutilityDataLoggerRegister (2110)

NV Member	Requirement	CNP Type	Direction	Description
1 "Register value output" (nvoRegVal)	Mandatory	SNVT_reg_val_ts	output	This network variable contains the current value of the register with a time stamp and status bits all contained within one variab
2 "Register historical period selection input" (nviHistChoice)	Optional	SNVT_count	input	This input network variable controls which history value is shown on the output network variable side via the network variables voHistVal and nvoHistTime.
3 "Register historical time selection input" (nviHistTime)	Optional	SNVT_time_stamp	input	This input network variable controls, according to the time, which history value is shown on the output network variable side via the network variables nvoHistVal and nvoHistTime.
4 "Register value input" (nviRegVal)	Optional	SNVT_reg_val	input	A value can be sent to the register object via the network using this network variable
5 "Register state selection input" (nviRegState)	Optional	SNVT_lev_disc	input	The register can be activated or deactivated by using this variable. When a register is activated the register object measures into the register. If the register is deactivated it does not measure but it is read/write-able.
6 "Measuring period ending input" (nviEndPeriod)	Optional	SNVT_time_stamp	input	This input network variable controls the ending of a billing period. When the billing period is terminated, the current register value will be transferred to the next history register. The series register cannot use this property.
7 "Register historical value output" (nvoHistVal)	Optional	SNVT_reg_val_ts	output	This output network variable contains the register object's historical value of a chosen time or history period. The output value is selected using the nviHistChoice or nviHistTime variables.

NV Member	Requirement	CNP Type	Direction	Description
8 "Register historical time output" (nvoHistTime)	Optional	SNVT_time_stamp	output	This output network variable contains a billing period end time. The output value is selected using the nviHistChoice and nviHistTime variables.
9 "Register measuring period output" (nvoMeasPeriod)	Optional	SNVT_elapsed_tm	output	This output network variable contains the length of the measuring period. If used, the register object measures series or maximum value data. Otherwise the value is zero.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTregName applying to SFPTutilityDataLoggerRegister	Optional	none	not permitted	The name of a utility data logger register device
SCPThighLimit1 applying to SFPTutilityDataLoggerRegister	Optional	none	not permitted	The alarm high limit against which the value field of the output value is tested for alarm conditions
SCPTsndDelta applying to SFPTutilityDataLoggerRegister	Optional	none	not permitted	The minimum change required to force transmission of the output value
SCPTbaseValue applying to SFPTutilityDataLoggerRegister	Optional	none	not permitted	The base value (where to begin counting)

7.31 SFPTutilityMeter (2201)

NV Member	Requirement	CNP Type	Direction	Description
1 "Meter Value Output" (nvoMeterVal)	Mandatory	SNVT_reg_val_ts	output	This network variable contains the present value of the meter, i.e., the actual running value hown by the display on the meter
2 "Historical Value Output" (nvoHistVal)	Mandatory	SNVT_reg_val_ts	output	This network variable contains, by default, a copy of the valid meter value at the turn of the last month
3 "Historical-Time Selection Input" (nviHistTime)	Mandatory	SNVT_time_stamp	input	This input network variable controls which history value is shown on the output network variable: nvoHistVal

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTsndDelta applying to nvoMeterVal	Mandatory	none	not permitted	The minimum change required to force transmission of the output value
SCPTtimePeriod applying to nvoHistVal	Mandatory	none	not permitted	This input configuration network variable defines the period of time between transfer of a values to the historical register
SCPTlocation applying to SFPTutilityMeter	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTbaseValue applying to SFPTutilityMeter	Optional	modify when disabled	not permitted	The base value (where to begin counting)
SCPTpulseValue applying to SFPTutilityMeter	Optional	modify when disabled	not permitted	This configuration property is used to scale the raw pulse value to an energy-meter value
SCPTnumDigits applying to SFPTutilityMeter	Optional	modify when disabled	not permitted	This configuration property is used for setting the total number of digits on the meter

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTobjMajVer applying to SFPTutilityMeter	Optional	constant	not permitted	The major version number for the object
SCPTobjMinVer applying to SFPTutilityMeter	Optional	device specific, constant	not permitted	The minor version number for the object

7.32 SFPTlampActuator (3040)

NV Member	Requirement	CNP Type	Direction	Description
1 "Lamp input value" (nviLampValue)	Mandatory	SNVT_switch	input	Permits another device to pass data to the Lamp Actuator
2 "Lamp feedback output" (nvoLampValueFb)	Mandatory	SNVT_switch	output	State of the Lamp Actuator (ON or OFF) and the percentage level of intensity
3 "Running hours output" (nvoRunHours)	Optional	SNVT_elapsed_tm	output	Monitors the Lamp Actuator's running (ON) hours
4 "Energy counter output" (nvoEnergyCnt)	Optional	SNVT_elec_kwh	output	Monitors the consumed energy in kilowatt hours

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTlocation applying to SFPTlampActuator	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTinFbDly applying to nvoLampValueFb	Optional	none	not permitted	The time period after the last update, in a succession of changes to the input, before the feedback output is updated

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTdefOutput applying to nvoLampValueFb	Optional	none	not permitted	The level the sensor should adopt in certain default conditions
SCPTrunHrInit applying to nvoRunHours	Optional	none	not permitted	
SCPTrunHrAlarm applying to nvoRunHours	Optional	none	not permitted	
SCPTenergyCntInit applying to nvoEnergyCnt	Optional	none	not permitted	

7.33 SFPTisiLampActuator (3041)

NV Member	Requirement	CNP Type	Direction	Description
1 "Lamp input value" (nviValue)	Mandatory	SNVT_switch_2	input	Scene, setting, occupancy, and/or request input to the lamp.
2 "Lamp or appliance feedback output" (nvoValueFb)	Mandatory	SNVT_switch_2	output	Reports the last requested scene, current setting, and power consumption for the lamp or appliance. Power consumption may be measured or estimated.
8 "Running hours output" (nvoRunHours)	Optional	SNVT_elapsed_tm	output	Reports total running hours
6 "Electrical energy upper portion" (nvoEnergyHi)	Optional	SNVT_elec_kwh	output	The nvoEnergyHi reading is incremented and the nvoEnergyLo reading wraps to zero when 1 kWh is reached for the nvoEnergyLo value. May be an estimated value.
4 "Power" (nvoPower)	Mandatory	SNVT_power	output	Reports power consumption. May be an estimate.

NV Member	Requirement	CNP Type	Direction	Description
5 "Color" (nviColor)	Optional	SNVT_color_2	input	
3 "Occupancy" (nvoOccupancyFb)	Mandatory	SNVT_occupancy	output	Occupancy state feedback
7 "Multiplier" (nvoMultiplierFb)	Optional	SNVT_multiplier_s	output	Level multiplier feedback. Reset to one after 60 min.
9 "Color feedback" (nvoColorFb)	Optional	SNVT_color_2	output	Color setting feedback; required if nviColor is implemented
10 "Electrical energy lower portion" (nvoEnergyLo)	Optional	SNVT_elec_whr	output	The nvoEnergyHi reading is incremented and the nvoEnergyLo reading wraps to zero when 1 kWh is reached for the nvoEnergyLo value. May be an estimated value.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTname1 applying to SFPTisiLampActuator	Mandatory	none	not permitted	Name of the object that the CP applies to. This name can be used by optional user interface applications or display devices.
SCPTdefInput applying to nviValue	Optional	none	not permitted	Input value when updates are not received, at power-on reset, or when overridden.
SCPTinFbDly applying to SFPTisiLampActuator	Optional	none	not permitted	The time period after the last update in a succession of changes to the input, before the feedback output is updated
SCPTobjMajVer applying to SFPTisiLampActuator	Optional	constant	not permitted	Major version number for the associated functional block. See also cpFbMinVer.
SCPTobjMinVer applying to SFPTisiLampActuator	Optional	device specific, constant	not permitted	Minor version number for the associated functional block. See also cpFbMajVer.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTminSendTime applying to nvoValueFb	Optional	none	not permitted	The minimum period of time between consecutive transmissions of the current value
SCPTrunHrInit applying to nvoMultiplierFb	Optional	none	not permitted	The initial value of the running hours counter network variable
SCPTrunHrAlarm applying to nvoMultiplierFb	Optional	none	not permitted	The alarm threshold for the running hours counter
SCPTlocation applying to SFPTisiLampActuator	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTenergyCntInit applying to nviColor	Optional	none	not permitted	The initial value of the energy counter for the associated network variable
SCPTscene applying to SFPTisiLampActuator	Mandatory	none	required with dimension 3..100	Scene table defining the minimum required scene definition entries. May be used in combination with cpSceneTiming.
SCPTsceneTiming applying to SFPTisiLampActuator	Optional	none	required with dimension 3..100	Scene timing definition used to supplement a scene table created with cpScene. This CP array defines the optional scene table entries for the ISI profiles. When used, it shall be used in combination with a cpScene array.
SCPTname2 applying to SFPTisiLampActuator	Mandatory	none	not permitted	Part 2 of the name of the functional block to be used by optional user interface applications. Shall be used with SCPTname1 and may optionally be used with SCPTname3. This part is concatenated after part 1, and may optionally be followed by part 3. Shall be implemented as a configuration network variable.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTname3 applying to SFPTisiLampActuator	Optional	none	not permitted	Part 3 of the name of the functional block to be used by optional user interface applications. Shall be used with SCPTname1 and SCPTname2. This part, if present, is concatenated with parts 1 and 2. Shall be implemented as a configuration network variable.
SCPTmaxSendTime applying to nvoValueFb	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value. If this value is set to the invalid value, the heartbeat interval will be 2 min when occupied and 20 min when unoccupied--this is the default behaviour. Shall be implemented as a configuration network variable.
SCPTsndDelta applying to nvoEnergyHi	Optional	none	not permitted	The minimum change required to force transmission of the output value
SCPTsceneName applying to SFPTisiLampActuator	Optional	none	required with dimension 3..100	Name for a scene to be used by optional user interface applications. Used to create an array that supplements a scene table created with a SCPTscene array.
SCPTlightingGroupEnable applying to SFPTisiLampActuator	Optional	none	required with dimension 8	Bit masks to enable or disable up to 64 ISI lighting groups. Group 0 is not used. Groups may also be enabled or disabled using a SNVT_switch_2 update.
SCPTmeasurementInterval applying to SFPTisiLampActuator	Optional	device specific, manufacturer specific	not permitted	Time period used for a power measurement; may be used to calibrate a power or energy sensor
SCPTovrBehave applying to SFPTisiLampActuator	Optional	none	not permitted	This parameter is used to define the behaviour when an override request is received via the Node Object.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTovrValue applying to SFPTisiLampActuator	Optional	none	not permitted	The value to be used for the input when the functional block is overridden, and the behaviour is "SPECIFIED"; if there is no Node object, a non-invalid value for the cpOvrValue state field triggers the override to the specified value, and the normal value is restored when the cpOvrValue state field changes to the invalid value
SCPTmaxSendTime applying to nvoEnergyHi	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTminSendTime applying to nvoEnergyHi	Optional	none	not permitted	The minimum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoEnergyLo	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTminSendTime applying to nvoEnergyLo	Optional	none	not permitted	The minimum period of time between consecutive transmissions of the current value
SCPTsndDelta applying to nvoEnergyLo	Optional	none	not permitted	The minimum change required to force transmission of the output value
SCPTmaxSendTime applying to nviColor	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTminSendTime applying to nviColor	Optional	none	not permitted	The minimum period of time between consecutive transmissions of the current value
SCPTsndDelta applying to nviColor	Optional	none	not permitted	The minimum change required to force transmission of the output value
SCPTloadControlOffset applying to SFPTisiLampActuator	Optional	none	not permitted	Offsets to be used during standby (unoccupied state but home, or sleep mode) and demand-response modes

7.34 SFPTconstantLightController (3050)

NV Member	Requirement	CNP Type	Direction	Description
1 "Illumination level input value" (nviLuxLevel)	Mandatory	SNVT_lux	input	Ambient light level
2 "Setting input" (nviSetting)	Mandatory	SNVT_setting	input	Selects the operating mode, and adjusts the setpoint of the constant light controller
3 "Lamp value output" (nvoLampValue)	Mandatory	SNVT_switch	output	State for the lamp actuator (ON or OFF), and the percentage level of intensity
4 "Manual override input" (nviManOverride)	Optional	SNVT_switch	input	Provides enable of manual control for the lamp value output

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTluxSetpoint applying to SFPTconstantLightController	Mandatory	none	not permitted	Used to change the illumination level setpoint for the controller
SCPTlocation applying to SFPTconstantLightController	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTminSendTime applying to SFPTconstantLightController	Optional	none	not permitted	Minimum period between output NV transmissions (maximum transmission rate)
SCPTmaxSendTime applying to SFPTconstantLightController	Optional	none	not permitted	Maximum period of time that expires before the object automatically transmits the present value of the lux level output NV
SCPTstep applying to SFPTconstantLightController	Optional	none	not permitted	Used to determine the maximum step that the constant light controller is allowed to take to approach the target illumination level

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTminDeltaLevel applying to SFPTconstantLightController	Optional	none	not permitted	Amount by which the lamp value output shall change before the lamp value is transmitted
SCPTonOffHysteresis applying to SFPTconstantLightController	Optional	none	not permitted	Hysteresis for the illumination level setpoint. The hysteresis is used in AUTO mode to switch-off and -on the lamp
SCPTclOffDelay applying to SFPTconstantLightController	Optional	none	not permitted	Used to determine the delay after which the lamp value output is switched-off
SCPTclOnDelay applying to SFPTconstantLightController	Optional	none	not permitted	Used to determine the delay after which the lamp value output is switched-on
SCPTpowerupState applying to SFPTconstantLightController	Optional	none	not permitted	Used to determine the state (mode) of the constant light controller object after power-up or reset. The state ("function") can be either SET_ON or SET_OFF.

7.35 SFPToccupancyController (3071)

NV Member	Requirement	CNP Type	Direction	Description
1 "Occupancy status input value" (nviOccupancy)	Mandatory	SNVT_occupancy	input	Provides the occupancy status
2 "Lamp value output" (nvoLampValue)	Mandatory	SNVT_switch	output	State for the lamp actuator (ON or OFF), and the percentage level of intensity
3 "Secondary occupancy status input value" (nviSecondary)	Optional	SNVT_occupancy	input	Provides the occupancy status of a neighbouring area in order to provide low-level lighting around an occupied area for a feeling of security

NV Member	Requirement	CNP Type	Direction	Description
4 "Setting input (Auto/Off)" (nviSetting)	Optional	SNVT_setting	input	The mode can be either ON(AUTO) or OFF
5 "Manual override input" (nviManOverride)	Optional	SNVT_switch	input	Enables the local and manual control of the lamp value output
6 "Setting output" (nvoSetting)	Optional	SNVT_setting	output	Selects the operating mode for another controller (e.g., constant light controller)

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTlocation applying to SFPToccupancyController	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTholdTime applying to SFPToccupancyController	Optional	none	not permitted	Hold time for occupied state after there is no occupancy detected
SCPTprimeVal applying to SFPToccupancyController	Optional	none	not permitted	Default lamp value sent when the area is occupied
SCPTsecondVal applying to SFPToccupancyController	Optional	none	not permitted	Default lamp value sent when the neighbouring area is occupied

7.36 SFPTswitch (3200)

NV Member	Requirement	CNP Type	Direction	Description
1 "Switch output value" (nvoSwitch)	Mandatory	SNVT_switch	output	
2 "Switch feedback value" (nviSwitchFb)	Optional	SNVT_switch	input	
3 "Setting output value" (nvoSetting)	Optional	SNVT_setting	output	

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTlocation applying to SFPTswitch	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTminSendTime applying to SFPTswitch	Optional	none	not permitted	Minimum period between output NV transmissions (maximum transmission rate)
SCPTmaxSendTime applying to SFPTswitch	Optional	none	not permitted	Maximum period of time that expires before the object automatically transmits the present value of the lux level output NV
SCPTstepValue applying to SFPTswitch	Optional	none	not permitted	When up/down pushbuttons are used, this parameter can be used to adjust the total ramp time from 0 to 100 %
SCPTmaxOut applying to SFPTswitch	Optional	none	not permitted	Used to limit the maximum value of the Switch output

7.37 SFPTscenePanel (3250)

NV Member	Requirement	CNP Type	Direction	Description
1 "Scene trigger output" (nvoScene)	Mandatory	SNVT_scene	output	
2 "Scene number feedback" (nviSceneFb)	Optional	SNVT_scene	input	Provides feedback from other scene panels
3 "Master fade output control setting" (nvoFadeSetting)	Optional	SNVT_setting	output	The lighting scene can be adjusted up or down without calling up a new scene

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTlocation applying to SFPTscenePanel	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTsceneNmbr applying to SFPTscenePanel	Optional	none	not permitted	The number of the first scene for the panel, other numbers are subsequent
SCPTminSendTime applying to nvoFadeSetting	Optional	none	not permitted	The minimum period of time between consecutive transmissions of the current value
SCPTstepValue applying to nvoFadeSetting	Optional	none	not permitted	The step value for up/down ramps or fade control

7.38 SFPTsceneController (3251)

NV Member	Requirement	CNP Type	Direction	Description
1 "Scene trigger input" (nviScene)	Mandatory	SNVT_scene	input	Triggers a scene, or loads the scene preset memory with preset values
2 "Switch output" (nvoSwitch)	Mandatory	SNVT_switch	output	Switch output for an actuator
3 "Master fade input" (nviSetting)	Optional	SNVT_setting	input	Used to adjust the scene output. The output is adjusted relative to the stored preset value.
4 "Direct control input" (nviSwitch)	Optional	SNVT_switch	input	Direct control method for the switch output
5 "Scene configuration input" (nviScenCf)	Optional	SNVT_scene_cfg	input	Used to change scene setup, and to read stored scene values
6 "Scene configuration output" (nvoScenFb)	Optional	SNVT_scene_cfg	output	Scene configuration output, updated with the memory content of requested preset

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTfadeTime applying to nviScene	Optional	none	not permitted	Used to set fade time; if the time is set to 0, meaning "learn", present functionality is used
SCPTdelayTime applying to nviScene	Optional	none	not permitted	Used to set delay time; if the time is set to 0, meaning "learn", present functionality is used
SCPTminSendTime applying to SFPTsceneController	Optional	none	not permitted	Minimum period between output NV transmissions (maximum transmission rate)

7.39 SFPTpartitionWallController (3252)

NV Member	Requirement	CNP Type	Direction	Description
1 "Primary-side scene input" (nviScene1)	Mandatory	SNVT_scene	input	Reads in data from the scene panels and other sensors of the primary side of the partition wall
2 "Primary-side scene output" (nvoScene1)	Mandatory	SNVT_scene	output	Provides the scene output to the scene controllers on the primary side of the partition wall
3 "Partition wall switch input" (nviWallSwitch)	Mandatory	SNVT_switch	input	The valid range of the input is open (state ON) and closed (state OFF). Other enumerations of the state are discarded; value field has no effect.
4 "Secondary-side scene output" (nvoScene2)	Mandatory	SNVT_scene	output	Provides the scene output to the scene controllers on the secondary side of the partition wall
5 "Secondary-side scene input" (nviScene2)	Mandatory	SNVT_scene	input	Reads in data from the scene panels and other sensors of the secondary side of the partition wall
6 "Primary-side fade input" (nviFade1)	Optional	SNVT_setting	input	Reads in data from the scene panels and other sensors of the primary side of the partition wall
7 "Primary-side fade output" (nvoFade1)	Optional	SNVT_setting	output	Provides SNVT_setting type output to the scene controllers on the primary side of the partition wall
8 "Secondary-side fade input" (nviFade2)	Optional	SNVT_setting	input	Reads in data from the scene panels and other sensors of the secondary side of the partition wall
9 "Secondary-side fade output" (nvoFade2)	Optional	SNVT_setting	output	Provides SNVT_setting type output to the scene controllers on the secondary side of the partition wall

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTlocation applying to SFPTpartitionWallController	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTsceneOffset applying to SFPTpartitionWallController	Optional	none	not permitted	The offset for the scene number when data is forwarded from primary to secondary

7.40 SFPTisiKeypad (3253)

NV Member	Requirement	CNP Type	Direction	Description
2 "Switch output value" (nvoSwitch)	Mandatory	SNVT_switch_2	output	Selected state, scene, and setting output. Also reports power consumption by the keypad, which may be measured or estimated.
1 "Switch feedback input" (nviSwitchFb)	Optional	SNVT_switch_2	input	Optional feedback input that may be used to synchronize the keypad state to other devices such as other keypads and lamps.
3 "Running hours output" (nvoRunHours)	Optional	SNVT_elapsed_tm	output	Reports total running hours

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTname1 applying to SFPTisiKeypad	Mandatory	none	not permitted	Part 1 of the name of the functional block to be used by optional user interface applications. May optionally used with SCPTname2 and SCPTname3. Shall be implemented as a configuration network variable.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTname2 applying to SFPTisiKeypad	Mandatory	none	not permitted	Part 2 of the name of the functional block to be used by optional user interface applications. Shall be used with SCPTname1 and may optionally be used with SCPTname3. This part is concatenated after part 1, and may optionally be followed by part 3. Shall be implemented as a configuration network variable.
SCPTname3 applying to SFPTisiKeypad	Optional	none	not permitted	Part 3 of the name of the functional block to be used by optional user interface applications. Shall be used with SCPTname1 and SCPTname2. This part, if present, is concatenated with parts 1 and 2. Shall be implemented as a configuration network variable.
SCPTlocation applying to SFPTisiKeypad	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTbuttonPressAction applying to SFPTisiKeypad	Mandatory	none	permitted with dimension 0..254	Button action definition used to create a button pressed action array, with an entry per button. This SCPT defines the minimum entries required by the ISI profiles.
SCPTbuttonColor applying to SFPTisiKeypad	Optional	none	not permitted	Button colour configuration for on and off states of a button. May be used to create an array that is used with a SCPTbuttonAction array to specify keypad button behaviour.
SCPTminSendTime applying to SFPTisiKeypad	Optional	none	not permitted	The minimum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nviSwitchFb	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value. If this value is set to the invalid value, the heartbeat interval will be 2 min when occupied and 20 min when unoccupied -- this is the default behaviour. Shall be implemented as a configuration network variable.
SCPTdefOutput applying to SFPTisiKeypad	Optional	none	not permitted	Output value at power-on reset or when overridden

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTinFbDly applying to SFPTisiKeypad	Optional	none	not permitted	The time period after the last update in a succession of changes to the input, before the feedback output is updated
SCPTobjMajVer applying to SFPTisiKeypad	Optional	constant	not permitted	Major version number for the associated functional block. See also cpFbMinVer.
SCPTobjMinVer applying to SFPTisiKeypad	Optional	device specific, constant	not permitted	Minor version number for the associated functional block. See also cpFbMajVer.
SCPTrunHrInit applying to nvoRunHours	Optional	device specific, constant, modify when disabled	not permitted	The initial value of the running hours counter network variable
SCPTrunHrAlarm applying to nvoRunHours	Optional	modify when disabled	not permitted	The alarm threshold for the running hours counter
SCPTbuttonRepeatInterval applying to SFPTisiKeypad	Optional	none	not permitted	Time between updates when a button is held down. The updates themselves may be throttled by the application or a SCPTminSendTime CP. Used to create an array used with a SCPTbuttonAction CP array.
SCPTsceneName applying to SFPTisiKeypad	Optional	none	required with dimension 3..100	Name for a scene to be used by optional user interface applications. Used to create an array that supplements a scene table created with a SCPTscene array.
SCPTlightingGroupEnable applying to SFPTisiKeypad	Optional	none	required with dimension 8	Bit masks to enable or disable up to 64 ISI lighting groups. Group 0 is not used. Groups may also be enabled or disabled using a SNVT_switch_2 update.

7.41 SFPTrealTimeKeeper (3300)

NV Member	Requirement	CNP Type	Direction	Description
1 "Real time output to the network" (nvoTimeDate)	Mandatory	SNVT_time_stamp	output	Actual real time and date on the network for all linked objects
2 "Time input from a synchronizing source" (nviTimeSet)	Optional	SNVT_time_stamp	input	To set the initial/seed time of the clock (for a clock that does not set its own initial time)

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmasterSlave applying to SFPTrealTimeKeeper	Optional	none	not permitted	"1" means "Master Clock Mode". "0" means "Slave Clock Mode".
SCPTupdateRate applying to nvoTimeDate	Optional	none	not permitted	Update rate of the output should be kept at 1 min, 1 h, or 1 day
SCPTsummerTime applying to SFPTrealTimeKeeper	Optional	none	not permitted	When not used entry shall be all "0" values. Year, minutes, and seconds are always "0".
SCPTwinterTime applying to SFPTrealTimeKeeper	Optional	none	not permitted	When not used entry shall be all "0" values. Year, minutes, and seconds are always "0".
SCPTmanualAllowed applying to SFPTrealTimeKeeper	Optional	none	not permitted	"1" means "Allowed". "0" means "Disallowed".

7.42 SFPTrealTimeBasedScheduler (3301)

NV Member	Requirement	CNP Type	Direction	Description
1 "Time event output" (nvoScene)	Mandatory	SNVT_scene	output	Provides a time- and date-related event for the controller
2 "Mode output" (nvoSetting)	Optional	SNVT_setting	output	Provides real-time-related operational-mode information for the controller

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTtimeEvent applying to SFPTrealTimeBasedScheduler	Mandatory	none	required with dimension 2..65535	Event or mode definitions to be transmitted if the time in the record is reached
SCPTdayDateIndex applying to SFPTrealTimeBasedScheduler	Mandatory	none	required with dimension 2..65535	One or two dates for matching with a start index to the time-event array
SCPTdefWeekMask applying to SFPTrealTimeBasedScheduler	Mandatory	none	not permitted	Day type definition for every day of the week
SCPTmodeHrtBt applying to nvoSetting	Optional	none	not permitted	The time that shall pass without an update for mode definitions to be automatically retransmitted, zero disables

7.43 SFPTlightingPanelController (3401)

NV Member	Requirement	CNP Type	Direction	Description
1 "Lighting Group Trigger" (nviGroup)	Mandatory	SNVT_scene	input	This input network variable triggers a group or pattern of lighting circuits.
2 "Lighting Group Feedback Output" (nvoGroupFb)	Mandatory	SNVT_scene	output	This output network variable provides current status of any of the possible 255 groups or patterns programmed into the controller.
3 "LPC Configuration Input" (nviLPCcfg)	Optional	SNVT_preset	input	This input network variable provides the functions required for programming groups, patterns and other lighting behaviours into the controller.
4 "LPC Configuration Feedback Output" (nvoLPCfb)	Optional	SNVT_preset	output	This output network variable provides information on current LPC configuration.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTlocation applying to SFPTlightingPanel Controller	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTmanualAllowed applying to SFPTlightingPanel Controller	Optional	none	not permitted	Provides a clock, with a manual time input, the possibility to permit manual time updating

7.44 SFPToutdoorLuminairController (3512)

NV Member	Requirement	CNP Type	Direction	Description
1 "Lamp Value Input" (nviLampValue)	Mandatory	SNVT_switch	input	The desired state (on/off) and value (%) of the lamp bulb
2 "Run-Hours Change" (nviRunHours)	Mandatory	SNVT_time_hour_p	input	The method used to reset or initialize the run-hours count

NV Member	Requirement	CNP Type	Direction	Description
3 "Electricity-Used Change" (nviEnergy)	Mandatory	SNVT_elec_kwh_l	input	The method used to reset or initialize the electricity-consumption total
4 "Lamp Value Output" (nvoLampValueFb)	Mandatory	SNVT_switch	output	The actual state (on/off) and value (%) of the lamp bulb
5 "Lamp-Controller Status Output" (nvoOLCStatus)	Mandatory	SNVT_lamp_status	output	The data related to the internal status conditions of the lamp controller
6 "OLC Environment Output" (nvoEnvironment)	Mandatory	SNVT_environment	output	The measured values from in and around the environment of the fixture, including mains/lamp voltages, mains/lamp currents, lamp temperature, consumed power, and power factor

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTminSendTime applying to SFPToutdoorLuminairController	Mandatory	none	not permitted	The minimum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to SFPToutdoorLuminairController	Mandatory	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTbkupSchedule applying to SFPToutdoorLuminairController	Mandatory	none	not permitted	The On/Off schedule to be used as a default
SCPTlampPower applying to SFPToutdoorLuminairController	Mandatory	none	not permitted	The maximum wattage of the installed bulbs.
SCPTenableStatusMsg applying to nvoOLCStatus	Mandatory	none	not permitted	Determines which status information will be available and valid for the alarm fields of nvoOLCStatus
SCPTdeviceOutSelection applying to SFPToutdoorLuminairController	Mandatory	none	not permitted	The type of hardware output
SCPTminSetpoint applying to SFPToutdoorLuminairController	Mandatory	none	not permitted	The minimum percentage value for a dimmable lamp to remain illuminated
SCPTrampUpTm applying to SFPToutdoorLuminairController	Mandatory	none	not permitted	The time it takes the lamp to warm-up to the full

CNP Type	Requirement	Restrictions	Arrays	Description
				operating temperature
SCPTmaxLevelVolt applying to SFPToutdoorLuminairController	Mandatory	none	not permitted	The maximum voltage (of the 1-10 V output) needed to achieve 100 % lamp-light output. In many cases, 100 % is reached before the 10 V-limit is reached
SCPTOLCLimits applying to nvoOLCStatus	Mandatory	none	not permitted	This configuration property sets the limits for nvoOLCStatus
SCPTpwrUpState applying to SFPToutdoorLuminairController	Mandatory	none	not permitted	This configuration property is be used to define the default output value on power up

7.45 SFPTidentifierSensor (5035)

NV Member	Requirement	CNP Type	Direction	Description
1 "ISO 7811" (nvoId)	Mandatory	SNVT_magcard	output	This data item contains data according to the ISO 7811, standard for card stripes.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to nvoId	Mandatory	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTlocation applying to SFPTidentifierSensor	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTobjMajVer applying to SFPTidentifierSensor	Optional	constant	not permitted	The major version number for the object

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTobjMinVer applying to SFPTidentifierSensor	Optional	constant	not permitted	The minor version number for the object
SCPTnwrkCnfg applying to SFPTidentifierSensor	Optional	reset after modification	not permitted	The value of this field determines the source of the node's network configuration.

7.46 SFPTentryExit (5051)

NV Member	Requirement	CNP Type	Direction	Description
1 "Entry state" (nviEntryState)	Mandatory	SNVT_ent_state	input	Desired state for an entry object, e.g. a door, lock, sluice, or something that controls entry of an area
2 "Entry status" (nvoEntryStatus)	Mandatory	SNVT_ent_status	output	Status information from an entry object, e.g., a door, lock, sluice, or something that allows/prohibits entry of an area
3 "Entry operation mode" (nviEntryOpMode)	Optional	SNVT_ent_opmode	input	Operation-mode for an entry object, e.g. a door, lock, sluice, or something which allows/prohibits entry to an area
4 "Warning alarm" (nvoPreAlarm)	Optional	SNVT_switch	output	Warning alarm setting when the device has been unable to perform a requested command longer than the nciPreAlarmDly time, but not as long as the nciAftAlarmDly time
5 "In lockable position" (nviPositioned)	Optional	SNVT_switch	input	Whether the mechanical equipment being subject to locking is in lockable position or not. Typically, this can be an open or closed door.
6 "After-warning alarm" (nvoAftAlarm)	Optional	SNVT_switch	output	Alarm setting when the device has been unable to perform a requested command for more than the nciPreAlarmDly time

NV Member	Requirement	CNP Type	Direction	Description
7 "Device-open request" (nviOpen)	Optional	SNVT_switch	input	opens the device in any state but EM_BLOCKED or EM_MANUAL. After the open command is taken away, the device goes back to its previous state.
8 "Device-open command feedback" (nvoOpenFb)	Optional	SNVT_switch	output	Feedback of the nviOpen network variable input
9 "Direction-flow control" (nviDirection)	Optional	SNVT_flow_dir	input	Control the direction of persons/items/fluids through device
10 "Direction-flow control feedback" (nvoDirectionFb)	Optional	SNVT_flow_dir	output	Direction of persons/items/fluids through device
11 "Service-message update request" (nviUpdateServ)	Optional	SNVT_switch	input	Request an update of an optional service-message output
12 "Service message" (nvoServiceMsg)	Optional	SNVT_str_asc	output	Current service-message information
13 "Sluice token input" (nviSluiceLock)	Optional	SNVT_switch	input	The sluice token input shall be set to TOKEN to indicate that an opening cycle is legal in case of sluice operation
14 "Sluice token feedback" (nvoSluiceLock)	Optional	SNVT_switch	output	Token feedback output in a sluice-lock configuration

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxRcvTime applying to SFPTentryExit	Mandatory	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to nvoEntryStatus	Mandatory	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTminSendTime applying to nvoEntryStatus	Mandatory	none	not permitted	The minimum period of time between consecutive transmissions of the current value
SCPTholdTime applying to SFPTentryExit	Optional	none	not permitted	Used to control the time the device is in open state before it returns to its former state (closed, closed/locked, etc.)
SCPTdefOutput applying to nviEntryState	Optional	none	not permitted	The position or level the actuator should adopt when updates are not received, or at power-on reset, or when overridden
SCPTdefOutput applying to nviEntryOpMode	Optional	none	not permitted	The position or level the actuator should adopt when updates are not received, or at power-on reset, or when overridden
SCPTdelayTime applying to nvoPreAlarm	Optional	none	not permitted	Delay time before a pre-alarm (warning) is sent
SCPTdelayTime applying to nvoAftAlarm	Optional	none	not permitted	Delay time before an alarm is sent
SCPTsafExtCnfg applying to nviEntryOpMode	Optional	none	not permitted	Mode that device shall be brought to for safety external state of operation
SCPTemergCnfg applying to nviEntryOpMode	Optional	none	not permitted	Mode that device shall be brought to for emergency state of operation
SCPTlocation applying to SFPTentryExit	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTobjMajVer applying to SFPTentryExit	Optional	constant	not permitted	The major version number for the object
SCPTobjMinVer applying to SFPTentryExit	Optional	device specific, constant	not permitted	The minor version number for the object

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTnwrkCnfg applying to SFPTentryExit	Optional	reset after modification	not permitted	The value of this field determines the source of the node's network configuration.
SCPTsluiceCnfg applying to SFPTentryExit	Optional	none	not permitted	Role of a device in a sluice operation

7.47 SFPTmodemController (5091)

NV Member	Requirement	CNP Type	Direction	Description
1 "Dial String Input" (nviDialStr)	Mandatory	SNVT_str_asc	input	Character string (30 characters max)
2 "Call Cancel Input" (nviCallCancel)	Mandatory	SNVT_switch	input	
3 "Connect Status Output" (nvoConnectStat)	Mandatory	SNVT_telcom	output	
4 "Connect String Output" (nvoConnectStr)	Optional	SNVT_str_asc	output	Character string (30 characters max)

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTautoAnswer applying to SFPTmodemController	Optional	none	not permitted	Enable the automatic call answer function of a device

7.48 SFPTtelephoneDirectory (5092)

NV Member	Requirement	CNP Type	Direction	Description
1 "Request Dial String Input" (nviReqDialStr)	Mandatory	SNVT_char_ascii	input	Telephone-number entry index (0-to-maximum), where maximum is less than or equal to 255
2 "Dial String Output" (nvoDialStr)	Mandatory	SNVT_str_asc	output	Character string (30 characters max)

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTdialString applying to SFPTtelephoneDirectory	Mandatory	none	permitted with dimension 0..256	Telephone number string used in dialling, including characters used for control

7.49 SFPTvariableSpeedMotorDrive (6010)

NV Member	Requirement	CNP Type	Direction	Description
1 "Drive speed setpoint" (nviDrvSpeedStpt)	Mandatory	SNVT_switch	input	Provides start/stop control and a low-resolution speed setpoint
2 "Drive speed setpoint scaling" (nviDrvSpeedScale)	Optional	SNVT_lev_percent	input	Negative values indicate a motor direction in reverse
3 "Drive current output" (nvoDrvCurnt)	Optional	SNVT_amp	output	Drive current (Amperes) output
4 "Drive speed output" (nvoDrvSpeed)	Mandatory	SNVT_lev_percent	output	Drive speed as a percentage of nominal speed

NV Member	Requirement	CNP Type	Direction	Description
5 "Drive voltage output" (nvoDrvVolt)	Optional	SNVT_volt	output	The typical range is 0 to 700 volts
6 "Drive power output" (nvoDrvPwr)	Optional	SNVT_power_kilo	output	Drive power (kW) output
7 "Drive total running hours" (nvoDrvRunHours)	Optional	SNVT_time_hour	output	The number of hours the drive has been actively running

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSetpoint applying to SFPTvariableSpeedMotorDrive	Mandatory	none	not permitted	The value is entered as a percent of nominal speed in RPM, as defined by the nominal speed configuration value
SCPTminSetpoint applying to SFPTvariableSpeedMotorDrive	Mandatory	none	not permitted	The value is entered as a percent of nominal speed in RPM, as defined by the nominal speed configuration value
SCPTmaxSendTime applying to SFPTvariableSpeedMotorDrive	Mandatory	none	not permitted	Controls the maximum period of time before the output values are transmitted
SCPTnomRPM applying to SFPTvariableSpeedMotorDrive	Mandatory	none	not permitted	This value is necessary to determine the minimum and maximum speed for the motor
SCPTnomFreq applying to SFPTvariableSpeedMotorDrive	Mandatory	none	not permitted	Typical default values include 50 Hz and 60 Hz
SCPTrampUpTm applying to SFPTvariableSpeedMotorDrive	Mandatory	none	not permitted	Minimum time of acceleration

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTrampDownTm applying to SFPTvariableSpeedMotorDrive	Mandatory	none	not permitted	Minimum time of deceleration
SCPTmaxRcvTime applying to SFPTvariableSpeedMotorDrive	Optional	none	not permitted	Maximum time that elapses after the last update to the input NV before actuator adopts the default output
SCPTminSendTime applying to SFPTvariableSpeedMotorDrive	Optional	none	not permitted	Controls the minimum period of time between output value transmissions
SCPTlocation applying to SFPTvariableSpeedMotorDrive	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTdefScale applying to nviDrvSpeedScale	Optional	none	not permitted	Default value for drive speed setpoint scaling

7.50 SFPTsunblindActuator (6110)

NV Member	Requirement	CNP Type	Direction	Description
1 "Sunblind Setting input" (nviSblndSet)	Mandatory	SNVT_setting	input	This input network variable is used to send the sunblind to a desired position
2 "Sunblind Control output for daisy chaining" (nvoSblndSetFwd)	Optional	SNVT_setting	output	This output network variable is used to provide feedback or to forward the input NV of nviSblndSet to another device or functional block
3 "Sunblind Status input" (nviSblndStatus)	Optional	SNVT_sblnd_state	input	This input network variable provides for receiving a Sunblind Controller status in order to report, via the Status output NV, the Sunblind Actuator status in conjunction with the Sunblind Controller status

NV Member	Requirement	CNP Type	Direction	Description
4 "Sunblind Override input" (nviSblndOvr)	Optional	SNVT_setting	input	This input network variable is used to send the sunblind to a desired position
5 "Sunblind Status output" (nvoSblndStatus)	Optional	SNVT_sblnd_state	output	This output network variable is used to provide feedback as to the actual sunblind position, error messages, and the cause of the latest change of the setpoint
6 "Sunblind feedback output for switch LEDs or general monitoring" (nvoMode)	Optional	SNVT_switch	output	This output network variable is used to provide a feedback output for switch LEDs or general monitoring.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to nvoSblndStatus	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoMode	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTlocation applying to SFPTsunblindActuator	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTobjMajVer applying to SFPTsunblindActuator	Optional	constant	not permitted	The major version number for the object
SCPTobjMinVer applying to SFPTsunblindActuator	Optional	device specific, constant	not permitted	The minor version number for the object

7.51 SFPTsunblindController (6111)

NV Member	Requirement	CNP Type	Direction	Description
1 "Controller setpoint output" (nvoSblndSetting)	Mandatory	SNVT_setting	output	This output network variable provides the Sunblind Controller setpoint value which may depend on any network input and configuration properties
2 "Sunblind controller state output" (nvoSblndState)	Optional	SNVT_sblnd_state	output	This output network variable is used to report the actual setpoint, error messages and the cause of the latest change of this setpoint
3 "Local setpoint adjustment" (nviLocalControl)	Optional	SNVT_setting	input	This network variable input is provided to set the controller setpoint output. Usually this command is given by a local control device
4 "Input for setpoint adjustment in groups" (nviGroupControl)	Optional	SNVT_setting	input	This network variable input is provided to set the controller setpoint output. Usually this command is given by a device which is intended to control groups of controllers or actuators
5 "Wind speed sensor input" (nviWindspeed)	Optional	SNVT_speed	input	This input network variable is used to get wind speed influence on the controller
6 "Outdoor brightness input standard range" (nviSunLux)	Optional	SNVT_lux	input	This input network variable is used to get sun (outdoor) brightness influence with a range from 0…65klux on the controller
7 "Rain sensor input" (nviRain)	Optional	SNVT_switch	input	This input network variable is used to get rain sensor influence on the controller
8 "Frost sensor input" (nviFrost)	Optional	SNVT_switch	input	This input network variable is used to get frost sensor influence on the controller
9 "Dawn state input" (nviDawn)	Optional	SNVT_switch	input	The term "Dawn" means the time before sunrise when it is more bright than during the night but not as bright as the average of daytime

NV Member	Requirement	CNP Type	Direction	Description
10 "Dusk state input" (nviDusk)	Optional	SNVT_switch	input	The term "Dusk" means the time before sunset when it is no longer as bright as the average of daytime but brighter than during the night
11 "Outdoor air temperature input" (nviOutdoorTemp)	Optional	SNVT_temp_p	input	This input network variable is used to get outdoor temperature sensor influence on the controller
12 "Indoor temperature input" (nviIndoorTemp)	Optional	SNVT_temp_p	input	This input network variable is used to get indoor temperature sensor influence on the controller
13 "Outdoor relative humidity input" (nviOutdoorRH)	Optional	SNVT_lev_percent	input	This input network variable is used to get outdoor relative humidity sensor influence on the controller
14 "Indoor relative humidity input" (nviIndoorRH)	Optional	SNVT_lev_percent	input	This input network variable is used to get indoor relative humidity sensor influence on the controller
15 "Indoor illumination level input" (nviIllumLev)	Optional	SNVT_lux	input	This input network variable is used to get indoor light sensor influence on the controller
16 "Scene trigger input" (nviScene)	Optional	SNVT_scene	input	Every scene relates to a particular setpoint value, which could be sent via nvoSblndSetting
17 "Global setpoint adjustment" (nviGlobalControl)	Optional	SNVT_setting	input	This network variable input is provided to set the controller setpoint output
18 "Window contact input" (nviWindowContact)	Optional	SNVT_switch	input	This input network variable is used to get window contact influence on the controller
19 "Mode enabling/disabling input" (nviAutoMode)	Optional	SNVT_switch	input	This input network variable is used to get mode switch functionality on the controller

NV Member	Requirement	CNP Type	Direction	Description
20 "Override state input" (nviOverride)	Optional	SNVT_switch	input	This input network variable is used to get override influence on the controller
21 "Setpoint input for maintenance reasons" (nviMaintenance)	Optional	SNVT_switch	input	This input network variable is used to get maintenance signal influence on the controller
22 "Heating/cooling demand input" (nviTerminalLoad)	Optional	SNVT_lev_percent	input	This input is used to receive the current heating/cooling demand of the system which the sunblind controller shares in
23 "Occupancy sensor value input" (nviOccSensor)	Optional	SNVT_occupancy	input	This input network variable is used to get occupancy sensor influence on the controller
24 "Occupancy override input" (nviOccManCmd)	Optional	SNVT_occupancy	input	This input network variable is used to get business hour info influence on the controller
25 "Glare detecting sensor value input" (nviGlare)	Optional	SNVT_switch	input	This input network variable is used to get glare detecting sensor influence on the controller
26 "Astronomical sensor value input for sun declination" (nviSunElevation)	Optional	SNVT_angle_deg	input	This network input represents information from a sun-position calculating device
27 "Astronomical sensor value input for sun inclination" (nviSunAzimuth)	Optional	SNVT_angle_deg	input	This network input represents information from a sun-position calculating device
28 "Setpoint override input" (nviSetOverride)	Optional	SNVT_setting	input	This input network variable is used to get override influence on the controller

NV Member	Requirement	CNP Type	Direction	Description
29 "Setpoint input for maintenance reasons" (nviSetMaint)	Optional	SNVT_setting	input	This input network variable is used to get maintenance influence on the controller

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to nvoSblndSetting	Mandatory	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoSblndState	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxRcvTime applying to nviWindspeed	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviRain	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviFrost	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviWindowContact	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTnvPriority applying to SFPTsunblindController	Optional	none	required with dimension 2..255	Defines the priority of inputs that control the positioning of the sunblinds

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTbypassTime applying to SFPTsunblindController	Optional	none	not permitted	The maximum amount of time that the controller can be in the bypass (occupancy) mode following the last bypass request. Zero disables the timer.
SCPTdefaultSetting applying to SFPTsunblindController	Optional	none	not permitted	Defines the default position of the sunblind in the event that the sensor input fails
SCPTdefaultSetting applying to nviWindowContact	Optional	none	not permitted	Defines the default position of the sunblind in the event that the sensor input fails
SCPTlocation applying to SFPTsunblindController	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTobjMajVer applying to SFPTsunblindController	Optional	constant	not permitted	The major version number for the object
SCPTobjMinVer applying to SFPTsunblindController	Optional	device specific, constant	not permitted	The minor version number for the object

7.52 SFPTisiSunblindActuator (6112)

NV Member	Requirement	CNP Type	Direction	Description
1 "Sunblind input value" (nviValue)	Mandatory	SNVT_switch_2	input	Scene, setting and/or request input to the sunblind actuator.
2 "Sunblind or appliance feedback output." (nvoValueFb)	Mandatory	SNVT_switch_2	output	Reports the last requested scene and current setting.

NV Member	Requirement	CNP Type	Direction	Description
3 "Power" (nvoPower)	Optional	SNVT_power	output	Reports the current power level of the sunblind or appliance. Power level may be measured or estimated to detect blocked motors.
4 "Occupancy feedback" (nvoOccupancyFB)	Mandatory	SNVT_occupancy	output	
5 "Running hours output" (nvoRunHours)	Optional	SNVT_elapsed_tm	output	Reports total running hours
6 "Angular distance" (nvoRotation)	Optional	SNVT_angle_deg	output	Report the rotation angle

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTname1 applying to SFPTisiSunblindActuator	Mandatory	none	not permitted	Part 1 of the name of the functional block to be used by optional user interface applications. May optionally used with SCPTname2 and SCPTname3. Shall be implemented as a configuration network variable.
SCPTname2 applying to SFPTisiSunblindActuator	Mandatory	none	not permitted	Part 2 of the name of the functional block to be used by optional user interface applications. Shall be used with SCPTname1 and may optionally be used with SCPTname3. This part is concatenated after part 1, and may optionally be followed by part 3. Shall be implemented as a configuration network variable.
SCPTscene applying to SFPTisiSunblindActuator	Mandatory	none	required with dimension 3..100	Scene definition used to create a scene table. This SCPT defines the minimum entries required by the ISI profiles. May be used in combination with SCPTsceneTiming.
SCPTdefInput applying to nviValue	Optional	none	not permitted	Input value when updates are not received, at power-on reset, or when overridden.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTinFbDly applying to SFPTisiSunblindActuator	Optional	none	not permitted	The time period after the last update in a succession of changes to the input, before the feedback output is updated
SCPTobjMajVer applying to SFPTisiSunblindActuator	Optional	constant	not permitted	The major version number for the object
SCPTobjMinVer applying to SFPTisiSunblindActuator	Optional	device specific, constant	not permitted	The minor version number for the object
SCPTminSendTime applying to nvoValueFb	Optional	none	not permitted	The minimum period of time between consecutive transmissions of the current value
SCPTlocation applying to SFPTisiSunblindActuator	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTsceneTiming applying to SFPTisiSunblindActuator	Optional	none	required with dimension 3..100	Scene timing definition used to supplement a scene table created with a SCPTscene array. This SCPT defines the optional scene table entries for the ISI profiles. When used, it shall be used in combination with a SCPTscene array.
SCPTname3 applying to SFPTisiSunblindActuator	Optional	none	not permitted	Part 3 of the name of the functional block to be used by optional user interface applications. Shall be used with SCPTname1 and SCPTname2. This part, if present, is concatenated with parts 1 and 2. Shall be implemented as a configuration network variable.
SCPTmaxSendTime applying to nvoValueFb	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value. ISI heartbeats are specified if this value is set to the invalid value

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTsceneName applying to SFPTisiSunblindActuator	Optional	none	required with dimension 3..100	Name for a scene to be used by optional user interface applications. Used to create an array that supplements a scene table created with a SCPTscene array.
SCPTmaxPower applying to nvoPower	Optional	none	not permitted	Power level at which the sunblind actuator detects a blocked motor and switches off automatically.
SCPTlightingGroupEnable applying to SFPTisiSunblindActuator	Optional	none	required with dimension 8	Bit masks to enable or disable up to 64 ISI lighting groups. Group 0 is not used. Groups may also be enabled or disabled using a SNVT_switch_2 update.
SCPTloadControlOffset applying to SFPTisiSunblindActuator	Optional	none	not permitted	Offsets to be used during standby (unoccupied state but home, or sleep mode) and demand-response modes

7.53 SFPTvariableAirVolume (8010)

NV Member	Requirement	CNP Type	Direction	Description
1 "Space Temperature Input" (nviSpaceTemp)	Mandatory	SNVT_temp_p	input	This input network variable is used to connect an external space temperature sensor to the node.
2 "Temperature Setpoint Input" (nviSetPoint)	Mandatory	SNVT_temp_p	input	This input network variable is used to allow the temperature setpoint for the occupied and standby mode to be changed via network.
3 "Space Temperature Output" (nvoSpaceTemp)	Mandatory	SNVT_temp_p	output	This output network variable is used to send the value of the controlled space temperature to other nodes.
4 "Unit Status Output" (nvoUnitStatus)	Mandatory	SNVT_hvac_status	output	

NV Member	Requirement	CNP Type	Direction	Description
5 "Application Mode Input" (nviApplicMode)	Optional	SNVT_hvac_mode	input	This network variable input is used to coordinate the VAV object with the air handler control or any other supervisory controller or intelligent human interface device.
6 "VAV Manual Override Input" (nviManOverride)	Optional	SNVT_hvac_overid	input	This input network variable is used for commanding the controller into a manual mode.
7 "Setpoint Offset Input" (nviSetpointOffset)	Optional	SNVT_temp_p	input	This input network variable is used to shift the temperature setpoint via the network by adding nviSetpointOffset to the current setpoint.
8 "Occupancy Input" (nviOccCmd)	Optional	SNVT_occupancy	input	This input network variable is used to command the VAV object into different occupancy modes.
9 "Emergency Command Input" (nviEmergCmd)	Optional	SNVT_hvac_emerg	input	This input network variable is used to command the VAV object into different emergency modes.
10 "Box Flow Input" (nviBoxFlow)	Optional	SNVT_flow	input	This variable represents the input to a VAV controller object from a flow sensor on the network.
11 "Energy Hold-Off Input" (nviEnergyHoldOff)	Optional	SNVT_switch	input	This input is from a device such as a window contact sensor.
12 "Fan-Speed Command Input" (nviFanSpeedCmd)	Optional	SNVT_switch	input	This input network variable is used to connect an external fan speed switch to the node or to allow any supervisory device to override the fan speed controlled by the node's control algorithm.
13 "CO2 Sensor Input" (nviCO2)	Optional	SNVT_ppm	input	This input network variable measures the CO2 levels in PPM.
14 "Heater Override Input" (nviHeaterOverid)	Optional	SNVT_switch	input	This input disables to be turned on by the VAV controller.

NV Member	Requirement	CNP Type	Direction	Description
15 "Duct-Inlet Temperature Input" (nviDuctInTemp)	Optional	SNVT_temp_p	input	The duct inlet air temperature value could be obtained by the VAV controller object through this network variable.
16 "Effective Setpoint Output" (nvoEffectSetPt)	Optional	SNVT_temp_p	output	The output network variable is used to inform the effective setpoint temperature when the setpoint is changed by local means.
17 "Effective-Flow Control-Point Output" (nvoFlowControlPt)	Optional	SNVT_flow	output	The output network variable is used to inform the effective control point used by the flow control loop.
18 "Box Flow Output" (nvoBoxFlow)	Optional	SNVT_flow	output	The output network variable is used to provide the flow in the box.
19 "Terminal Load Output" (nvoTerminalLoad)	Optional	SNVT_lev_percent	output	The output network variable is used to provide the demand for supply energy.
20 "Energy Hold-Off Output" (nvoEnergyHoldOff)	Optional	SNVT_switch	output	This output is used to convey to other devices the state of an EnergyHoldOff device that is hardwired to the controller.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to SFPTvariableAirVolume	Mandatory	none	permitted with dimension 0..65500	The maximum period of time between consecutive transmissions of the current value
SCPTmaxRcvTime applying to SFPTvariableAirVolume	Optional	none	permitted with dimension 0..65500	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTminSendTime applying to SFPTvariableAirVolume	Optional	none	permitted with dimension 0..65500	The minimum period of time between consecutive transmissions of the current value

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTlocation applying to SFPTvariableAirVolume	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTductArea applying to SFPTvariableAirVolume	Optional	none	not permitted	The duct area used to calculate the air flow, relevant only for VAV actuators / controllers
SCPTminFlow applying to SFPTvariableAirVolume	Mandatory	none	not permitted	The minimum flow
SCPTmaxFlow applying to SFPTvariableAirVolume	Mandatory	none	not permitted	The maximum flow
SCPTminFlowHeat applying to SFPTvariableAirVolume	Optional	none	not permitted	The minimum airflow setpoint of a VAV terminal while heating
SCPTminFlowStby applying to SFPTvariableAirVolume	Optional	none	not permitted	The minimum flow through the VAV box in standby mode
SCPTnomAirFlow applying to SFPTvariableAirVolume	Optional	none	not permitted	Value used in calculating the air flow in an airflow control actuator
SCPTgainVAV applying to SFPTvariableAirVolume	Optional	none	not permitted	The gain of the VAV controller object
SCPTmaxSendTime applying to nvoBoxFlow	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoTerminalLoad	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to nvoEffectSetPt	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoEnergyHoldOff	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxRcvTime applying to nviSpaceTemp	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviSetpointOffset	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviApplicMode	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviBoxFlow	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviEnergyHoldOff	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviCO2	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviDuctInTemp	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTsetPnts applying to SFPTvariableAirVolume	Mandatory	none	not permitted	The occupancy temperature setpoints for heat and cool mode
SCPTmaxSendTime applying to nvoSpaceTemp	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoUnitStatus	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value

7.54 SFPTfanCoilUnit (8020)

NV Member	Requirement	CNP Type	Direction	Description
1 "Space temperature input" (nviSpaceTemp)	Mandatory	SNVT_temp_p	input	Optional input from a sensor node if the fan coil unit controller node itself provides a locally-wired space temperature sensor
2 "Temperature setpoint input" (nviSetPoint)	Mandatory	SNVT_temp_p	input	Allows the temperature setpoint for the occupied and standby mode to be changed via the network
3 "Heat control output" (nvoHeatOutput)	Mandatory	SNVT_lev_percent	output	Current position of the heat actuator; can be used as part of a control loop and for monitoring purposes
4 "Cool control output" (nvoCoolOutput)	Mandatory	SNVT_lev_percent	output	Current position of the cool actuator; can be used as part of a control loop and for monitoring purposes
5 "Fan speed output" (nvoFanSpeed)	Mandatory	SNVT_switch	output	Current fan speed of a multi-speed (n-speed) fan; can be used as part of a control loop and for monitoring purposes
6 "Fan speed command" (nviFanSpeedCmd)	Optional	SNVT_switch	input	Used to connect an external fan speed switch to the node

NV Member	Requirement	CNP Type	Direction	Description
7 "Occupancy input" (nviOccCmd)	Optional	SNVT_occupancy	input	Commands the fan coil unit controller object into different occupancy modes
8 "Application mode input" (nviApplicMode)	Optional	SNVT_hvac_mode	input	Coordinates this object with any supervisory controller providing the supply energy
9 "Setpoint offset input" (nviSetPtOffset)	Optional	SNVT_temp_p	input	This setpoint shifting operates only on occupied and standby setpoints. It does not affect the unoccupied setpoint.
10 "Water temperature input" (nviWaterTemp)	Optional	SNVT_temp_p	input	Automatic heat/cool switchover dependent on the supply temperature
11 "Terminal load output" (nvoTerminalLoad)	Optional	SNVT_lev_percent	output	Present heat/cool energy demand of the FCU controller
12 "Absolute power consumption output" (nvoLoadAbs)	Optional	SNVT_power	output	Present heat/cool power consumption of the FCU controller
13 "Discharge air temperature output" (nvoDischAirTemp)	Optional	SNVT_temp_p	output	Temperature of the air that leaves the fan coil
14 "Reheat output" (nvoReheat)	Optional	SNVT_switch	output	Present state of an multi-stage (n-stage) reheat output
15 "Space temperature output" (nvoSpaceTemp)	Optional	SNVT_temp_p	output	Space temperature, if this temperature is locally measured
16 "Effective setpoint output" (nvoEffectSetPt)	Optional	SNVT_temp_p	output	May depend on nciSetPnts, nviOccCmd, nviSetPoint, nviSetPtOffset, any local setpoint means

NV Member	Requirement	CNP Type	Direction	Description
17 "Discharge air temperature input" (nviDischAirTemp)	Optional	SNVT_temp_p	input	The network value takes precedence over the physical value if both are available.
18 "Energy hold-off input" (nviEnergyHoldOff)	Optional	SNVT_switch	input	Input from a device such as a window contact sensor
19 "Occupancy output" (nvoOccCmd)	Optional	SNVT_occupancy	output	May derive from an occupancy sensor or any other means affecting the occupancy mode
20 "Energy hold-off output" (nvoEnergyHoldOff)	Optional	SNVT_switch	output	Provides other devices the state of an EnergyHoldOff device that is hardwired to the controller
21 "Unit status output" (nvoUnitStatus)	Optional	SNVT_hvac_status	output	Combines operating mode, capacity of heating/cooling used, and an indication if any alarms are present in the object

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to SFPTfanCoilUnit	Mandatory	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTsetPnts applying to SFPTfanCoilUnit	Mandatory	none	not permitted	The occupancy temperature setpoints for heat and cool mode
SCPTmaxRcvTime applying to SFPTfanCoilUnit	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTminSendTime applying to SFPTfanCoilUnit	Optional	none	not permitted	The minimum period of time between consecutive transmissions of the current value
SCPTlocation applying to SFPTfanCoilUnit	Optional	none	not permitted	Provides descriptive physical location information related to the object.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTnumValves applying to SFPTfanCoilUnit	Optional	none	not permitted	The value 1 implies one output valve (two-pipe system) and the value 2 implies two output valves (four pipe system).

7.55 SFPTroofTopUnit (8030)

NV Member	Requirement	CNP Type	Direction	Description
1 "Space Temperature Input" (nviSpaceTemp)	Mandatory	SNVT_temp_p	input	This input network variable is used to connect an external space temperature sensor to the node.
2 "Temperature Setpoint Input" (nviSetPoint)	Mandatory	SNVT_temp_p	input	This input network variable is used to allow the temperature setpoint for the occupied and standby mode to be changed via the network.
3 "Space Temperature Output" (nvoSpaceTemp)	Mandatory	SNVT_temp_p	output	This output network variable is used to send the value of the controlled space temperature (either from a locally connected space temperature sensor or from the nviSpaceTemp input) to other nodes.
4 "Unit Status Output" (nvoUnitStatus)	Mandatory	SNVT_hvac_status	output	This output network variable is available to report the object status.
5 "Application Mode Input" (nviApplicMode)	Optional	SNVT_hvac_mode	input	This network variable input is used to coordinate the RTU Device with any supervisory controller or intelligent human interface device.
6 "Occupancy Input" (nviOccCmd)	Optional	SNVT_occupancy	input	This input network variable is used to command the RTU object into different occupancy modes. It is typically set by a supervisory node.
7 "Setpoint Offset Input" (nviSetPtOffset)	Optional	SNVT_temp_p	input	This input network variable is used to shift the temperature setpoint via the network by adding nviSetPtOffset to the current setpoint.

NV Member	Requirement	CNP Type	Direction	Description
8 "Outside Temperature Input" (nviOutsideTemp)	Optional	SNVT_temp_p	input	This input network variable is the current outside air temperature.
9 "Outside Humidity Input" (nviOutsideRH)	Optional	SNVT_lev_percent	input	This input network variable is the current outside air relative humidity.
10 "Effective Setpoint Output" (nvoEffectSetPt)	Optional	SNVT_temp_p	output	The output network variable is used to inform the effective setpoint temperature when the setpoint is changed by nciSetPnts, nviSetPoint, nviSetPointOffset or local means.
11 "Outside Temperature Output" (nvoOutsideTemp)	Optional	SNVT_temp_p	output	This output network variable is the current outside air temperature.
12 "Outside Humidity Output" (nvoOutsideRH)	Optional	SNVT_lev_percent	output	This output network variable is the current outside air relative humidity.
16 "CO2 Sensor Output" (nvoCO2)	Optional	SNVT_ppm	output	This output network variable is the CO2 measurement.
13 "Space Humidity Input" (nviSpaceRH)	Optional	SNVT_lev_percent	input	This input network variable measures the space relative humidity.
14 "CO2 Sensor Input" (nviCO2)	Optional	SNVT_ppm	input	This input network variable measures the CO2 levels in PPM.
15 "Emergency Command Input" (nviEmergCmd)	Optional	SNVT_hvac_emerg	input	This input network variable is used to command the RTU object into different emergency modes.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to nvoEffectSetPt	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoOutsideRH	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nviEmergCmd	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoOutsideTemp	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTsetPnts applying to SFPTroofTopUnit	Mandatory	none	not permitted	The occupancy temperature setpoints for heat and cool mode
SCPTmaxRcvTime applying to nviSpaceRH	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviSpaceTemp	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviSetPtOffset	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviOutsideTemp	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviOutsideRH	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxRcvTime applying to nviApplicMode	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nvoCO2	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTlocation applying to SFPTroofTopUnit	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTlimitCO2 applying to SFPTroofTopUnit	Optional	none	not permitted	CO2 threshold limit, controller to maintain concentration below this limit
SCPTmaxSendTime applying to SFPTroofTopUnit	Mandatory	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoSpaceTemp	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoUnitStatus	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxRcvTime applying to SFPTroofTopUnit	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables

7.56 SFPTchiller (8040)

NV Member	Requirement	CNP Type	Direction	Description
1 "Chiller enable" (nviChillerEnable)	Mandatory	SNVT_switch	input	Request start/stop chiller
2 "Cool setpoint" (nviCoolSetpt)	Mandatory	SNVT_temp_p	input	Desired temperature of leaving chilled water
3 "On or off state" (nvoOnOff)	Mandatory	SNVT_switch	output	Chiller on/off run state
4 "Active setpoint output" (nvoActiveSetpt)	Mandatory	SNVT_temp_p	output	Active cool or heat setpoint
5 "Capacity limit input" (nviCapacityLim)	Optional	SNVT_lev_percent	input	Capacity limit of chiller
6 "Entering chilled-water temperature input" (nviEntChWTemp)	Optional	SNVT_temp_p	input	Accommodates remote temperature-sensor input
7 "Chiller mode" (nviMode)	Optional	SNVT_hvac_mode	input	Chiller modes
8 "Heating setpoint" (nviHeatSetpt)	Optional	SNVT_temp_p	input	Heating setpoint
9 "Actual capacity level" (nvoActCapacity)	Optional	SNVT_lev_percent	output	Actual running capacity of unit
10 "Capacity limit" (nvoCapacityLim)	Optional	SNVT_lev_percent	output	Present capacity limit setting of chiller

NV Member	Requirement	CNP Type	Direction	Description
11 "Leaving chilled-water temperature" (nvoLvgChWTemp)	Optional	SNVT_temp_p	output	Leaving chilled-water temperature
12 "Entering chilled-water temperature" (nvoEntChWTemp)	Optional	SNVT_temp_p	output	Entering chilled-water temperature
13 "Entering condenser-water temperature" (nvoEntCndWTemp)	Optional	SNVT_temp_p	output	Entering condenser-water temperature
14 "Leaving condenser-water temperature" (nvoLvgCndWTemp)	Optional	SNVT_temp_p	output	Leaving condenser-water temperature
15 "Alarm annunciation text" (nvoAlarmDescr)	Optional	SNVT_str_asc	output	Character string (30 characters max)
16 "Chiller status" (nvoChillerStat)	Optional	SNVT_chlr_status	output	Chiller states or modes

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTpwrUpState applying to SFPTchiller	Mandatory	none	not permitted	The default power-up and restart modes of the device when the default behaviour selector is set to zero
SCPTmaxSendTime applying to SFPTchiller	Mandatory	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTminSendTime applying to SFPTchiller	Mandatory	none	not permitted	The minimum period of time between consecutive transmissions of the current value
SCPTlimitChlrCap applying to SFPTchiller	Optional	none	not permitted	The default value for the capacity limit of the chiller when the default behaviour selector is set to zero

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTcoolSetpt applying to SFPTchiller	Mandatory	none	not permitted	The default setpoint for the leaving chilled water temperature in cooling mode when the default behaviour selector is set to zero
SCPTheatSetpt applying to SFPTchiller	Optional	none	not permitted	The default setpoint for the leaving water temperature in heating mode when the default behaviour selector is set to zero
SCPThvacMode applying to SFPTchiller	Optional	none	not permitted	The default operating mode of the device when the default behaviour selector is set to zero
SCPTlocation applying to SFPTchiller	Optional	none	not permitted	Provides descriptive physical location information related to the object
SCPTdefltBehave applying to SFPTchiller	Optional	none	not permitted	Selects which set of values will be used on power-up and communication failure, between the stated default values (0), or manufacturer-specified values (1)
SCPTmaxRcvTime applying to SFPTchiller	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTpwrUpDelay applying to SFPTchiller	Optional	none	not permitted	The minimum period of time after power-up or re-establishment of communications before a control action takes place

7.57 SFPTheatPump (8051)

NV Member	Requirement	CNP Type	Direction	Description
1 "Remote Space Temperature" (nviSpaceTemp)	Mandatory	SNVT_temp_p	input	This input network variable is used to connect an external space temperature sensor to the node.
2 "Setpoint from Network" (nviSetPoint)	Mandatory	SNVT_temp_p	input	This input network variable is used to allow the temperature setpoint for the occupied mode to be changed via the network.
3 "Space Temp Value Used" (nvoSpaceTemp)	Mandatory	SNVT_temp_p	output	This output network variable is used to send the value of the controlled space temperature to other nodes.
4 "Current Status of Unit" (nvoUnitStatus)	Mandatory	SNVT_hvac_status	output	This output network variable is available to report the object status.
5 "Command from System" (nviApplicMode)	Optional	SNVT_hvac_mode	input	This network variable input is used to coordinate the Heat Pump with any supervisory controller or intelligent human interface device.
6 "HVAC Occupancy Status" (nviOccCmd)	Optional	SNVT_occupancy	input	This input network variable is used to place the heat pump into different occupancy modes. It is typically set by a supervisory node
7 "Simple Occupancy Status" (nviOccupSw)	Optional	SNVT_switch	input	This input network variable provides an indication of the simple two-state occupancy status of the unit, which is either unoccupied or occupied.
8 "Setpoint Offset" (nviSetPtOffset)	Optional	SNVT_temp_p	input	This input network variable specifies the direction and magnitude of the shift of the current occupied setpoints.
9 "Auxilliary Heat Enable" (nviAuxHeat)	Optional	SNVT_switch	input	This input network variable indicates whether auxiliary heat has been enabled or disabled.

NV Member	Requirement	CNP Type	Direction	Description
10 "Setpoint Value Used" (nvoEffectSetPt)	Optional	SNVT_temp_p	output	This input network variable provides an indication of the setpoint being used for control.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTsetPnts applying to SFPTheatPump	Mandatory	none	not permitted	The occupancy temperature setpoints for heat and cool mode
SCPTmaxSendTime applying to nvoEffectSetPt	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTlocation applying to SFPTheatPump	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTmaxRcvTime applying to nviSpaceTemp	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviSetPoint	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviApplicMode	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviOccCmd	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviOccupSw	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxRcvTime applying to nviSetPtOffset	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviAuxHeat	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxSendTime applying to SFPTheatPump	Mandatory	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoSpaceTemp	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoUnitStatus	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxRcvTime applying to SFPTheatPump	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables

7.58 SFPTthermostat (8060)

NV Member	Requirement	CNP Type	Direction	Description
1 "Temperature Setpoint Input" (nviSetPoint)	Mandatory	SNVT_temp_p	input	This input network variable is used to allow the temperature setpoint for the occupied and standby mode being changed via the network.
2 "Heat Control Output" (nvoHeatOutput)	Mandatory	SNVT_lev_percent	output	This output network variable reflects the current position of the heat actuator and can be used as part of a control loop and for monitoring purposes.

NV Member	Requirement	CNP Type	Direction	Description
3 "Cool Control Output" (nvoCoolOutput)	Mandatory	SNVT_lev_percent	output	This output network variable reflects the current position of the cool actuator and can be used as part of a control loop and for monitoring purposes.
4 "Space Temperature Output" (nvoSpaceTemp)	Mandatory	SNVT_temp_p	output	This output network variable is used to send the value of a locally connected space temperature sensor to other nodes. It is mandatory to the profile.
5 "Unit Status Output" (nvoUnitStatus)	Mandatory	SNVT_hvac_status	output	This output network variable is available to report the object status.
6 "Space Temperature Input" (nviSpaceTemp)	Optional	SNVT_temp_p	input	This input network variable is used to connect an external space temperature sensor to the node.
7 "Occupancy Input" (nviOccCmd)	Optional	SNVT_occupancy	input	This input network variable is used to command the thermostat object into different occupancy modes.
8 "Application Mode Input" (nviApplicMode)	Optional	SNVT_hvac_mode	input	This network variable input is used to coordinate the thermostat object with any supervisory controller providing the supply energy, e.g. hot or cold water.
9 "Setpoint Offset Input" (nviSetPtOffset)	Optional	SNVT_temp_p	input	This input network variable is used to shift the temperature setpoint via network by adding nviSetPtOffset to the current setpoint.
10 "Energy Hold Off Input" (nviEnergyHoldOff)	Optional	SNVT_switch	input	This input is from a device such as a door/window contact sensor.
11 "Terminal Load Output" (nvoTerminalLoad)	Optional	SNVT_lev_percent	output	This output network variable reflects the current heat/cool energy demand of the Thermostat Device which is typically bound to an energy providing node.

NV Member	Requirement	CNP Type	Direction	Description
12 "Effective Setpoint Output" (nvoEffectSetPt)	Optional	SNVT_temp_p	output	The output network variable is used to report the effective setpoint temperature when the setpoint is changed through nciSetPnts, nviSetPoint, nviSetPtOffset or by local means.
13 "Terminal Fan" (nvoTerminalFan)	Optional	SNVT_switch	output	This output network variable reflects the current fan speed of a multi-speed (n-speed) fan. It can be used as part of a control loop and for monitoring purposes.
14 "Energy Hold Off Output" (nvoEnergyHoldOff)	Optional	SNVT_switch	output	This output is used to convey to other devices the state of an EnergyHoldOff device that is hardwired to the controller. Refer to EnergyHoldOff Input.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTsetPnts applying to SFPTthermostat	Mandatory	none	not permitted	The occupancy temperature setpoints for heat and cool mode
SCPTmaxRcvTime applying to nviSpaceTemp	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviApplicMode	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviSetPtOffset	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviEnergyHoldOff	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTminDeltaTemp applying to SFPTthermostat	Optional	none	not permitted	The minimum change in temperature required to be treated as significant
SCPTlocation applying to SFPTthermostat	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTheatUpperSP applying to SFPTthermostat	Optional	none	not permitted	Limits the upper extent of the permitted range for the heating setpoint
SCPTheatLowerSP applying to SFPTthermostat	Optional	none	not permitted	Limits the lower extent of the permitted range for the heating setpoint
SCPTcoolUpperSP applying to SFPTthermostat	Optional	none	not permitted	Limits the upper extent of the permitted range for the cooling setpoint
SCPTcoolLowerSP applying to SFPTthermostat	Optional	none	not permitted	Limits the lower extent of the permitted range for the cooling setpoint
SCPTmaxSendTime applying to SFPTthermostat	Mandatory	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoHeatOutput	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoCoolOutput	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoSpaceTemp	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoUnitStatus	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxRcvTime applying to SFPTthermostat	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to nvoEnergyHoldOff	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoEffectSetPt	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value

7.59 SFPTchilledCeilingController (8070)

NV Member	Requirement	CNP Type	Direction	Description
1 "Space Temperature Input" (nviSpaceTemp)	Mandatory	SNVT_temp_p	input	This input network variable is used to connect an external space temperature sensor to the node.
2 "Temperature Setpoint Input (absolute)" (nviSetpoint)	Optional	SNVT_temp_p	input	This input network variable is used to allow the temperature setpoints for the occupied and standby modes to be changed via the network.
3 "Setpoint Offset Input" (nviSetptOffset)	Optional	SNVT_temp_p	input	This input network variable is used to shift the effective occupied and standby temperature setpoints by adding nviSetptOffset to the current setpoints.
4 "Setpoint Shift Input" (nviSetptShift)	Optional	SNVT_temp_setpt	input	This input network variable is used to shift the effective heat/cool setpoints by adding the corresponding value in nviSetptShift to the current setpoints.
5 "Occupancy Scheduler Input" (nviOccSchedule)	Optional	SNVT_tod_event	input	This input network variable is used to command the Chilled Ceiling Controller into different occupancy modes.
6 "Occupancy Override Input" (nviOccManCmd)	Optional	SNVT_occupancy	input	This input network variable is used to command the Chilled Ceiling Controller into different occupancy modes.

NV Member	Requirement	CNP Type	Direction	Description
7 "Occupancy Sensor Input" (nviOccSensor)	Optional	SNVT_occupancy	input	This input network variable is used to indicate the presence of occupants in the controlled space.
8 "Application Mode Input" (nviApplicMode)	Optional	SNVT_hvac_mode	input	This network variable input is used to coordinate the Chilled Ceiling Controller with any supervisory controller.
9 "Heat/Cool Mode Input" (nviHeatCool)	Optional	SNVT_hvac_mode	input	This network variable input is used to coordinate the Chilled Ceiling Controller with any node that may need to control the heat/cool changeover of the unit.
10 "Energy Hold Off Input" (nviEnergyHoldOff)	Optional	SNVT_switch	input	This input is used to stop heating and cooling while allowing the unit to protect the space from temperature extremes.
11 "Water Valve Override Input" (nviValveOverride)	Optional	SNVT_hvac_overid	input	This input network variable is used for commanding the controller into a manual mode for overriding water valves controlled by the unit.
12 "Source Temperature Input" (nviSourceTemp)	Optional	SNVT_temp_p	input	This input network variable is used to indicate the temperature of the air or water being supplied to the unit for heating and/or cooling capacity.
13 "Heat Source Temperature Input" (nviHeatSrcTemp)	Optional	SNVT_temp_p	input	This input network variable is used to indicate the temperature of the air or water being supplied to the unit for heating capacity.
14 "Cool Source Temperature Input" (nviCoolSrcTemp)	Optional	SNVT_temp_p	input	This input network variable is used to indicate the temperature of the air or water being supplied to the unit for cooling capacity.
15 "Space Humidity Input" (nviSpaceRH)	Optional	SNVT_lev_percent	input	This input network variable is the measured space humidity in percent. This input is typically sent from a communicating humidity sensor.
16 "Space Dew Point Temperature Input" (nviSpaceDewPt)	Optional	SNVT_temp_p	input	This input network variable is the measured space dew point temperature.

NV Member	Requirement	CNP Type	Direction	Description
17 "Outdoor Air Dew Point Temp. Input" (nviOutdoorDewPt)	Optional	SNVT_temp_p	input	This input network variable is the measured outdoor dew point temperature.
18 "Primary Heat Input for Slave Operation" (nviHeatPriSlave)	Optional	SNVT_lev_percent	input	This input network variable is intended for slave operation.
19 "Secondary Heat Input for Slave Operation" (nviHeatSecSlave)	Optional	SNVT_lev_percent	input	This input network variable is intended for slave operation.
20 "Primary Cool Input for Slave Operation" (nviCoolPriSlave)	Optional	SNVT_lev_percent	input	This input network variable is intended for slave operation.
21 "Secondary Cool Input for Slave Operation" (nviCoolSecSlave)	Optional	SNVT_lev_percent	input	This input network variable is intended for slave operation.
22 "Effective Space Temperature Output" (nvoSpaceTemp)	Mandatory	SNVT_temp_p	output	This output network variable is used to monitor the effective space temperature that the Chilled Ceiling Controller is using for control.
23 "Unit Status Output" (nvoUnitStatus)	Mandatory	SNVT_hvac_status	output	This output network variable is available to report the Chilled Ceiling Controller status.
24 "Effective Setpoint Output" (nvoEffectSetpt)	Optional	SNVT_temp_p	output	This output network variable is used to monitor the effective temperature setpoint.
25 "Effective Occupancy Output" (nvoEffectOccup)	Optional	SNVT_occupancy	output	This output network variable is used to indicate the actual occupancy mode of the unit.
26 "Effective Heat/Cool Output" (nvoHeatCool)	Optional	SNVT_hvac_mode	output	This output network variable is used to indicate the actual heat/cool mode of the unit.

NV Member	Requirement	CNP Type	Direction	Description
27 "Local Setpoint Output" (nvoSetpoint)	Optional	SNVT_temp_p	output	This output network variable is used to monitor the space temperature setpoint if a setpoint device is locally wired.
28 "Local Setpoint Shift Output" (nvoSetptShift)	Optional	SNVT_temp_setpt	output	This output network variable is used to report a locally-determined shift of the effective heat/cool setpoints.
29 "Absolute Power Consumption Output" (nvoLoadAbs)	Optional	SNVT_power	output	This output network variable can used to indicate the current power consumption of the unit.
30 "Terminal Load Output" (nvoTerminalLoad)	Optional	SNVT_lev_percent	output	This output indicates the current heat/cool energy demand of the unit.
31 "Primary Heat Output" (nvoHeatPrimary)	Optional	SNVT_lev_percent	output	This output network variable reflects the current level of the primary heat output or can be used to control a remote primary heat source.
32 "Secondary Heat Output" (nvoHeatSecondary)	Optional	SNVT_lev_percent	output	This output network variable reflects the current level of the secondary heat output or can be used to control a remote secondary heat source.
33 "Primary Cool Output" (nvoCoolPrimary)	Optional	SNVT_lev_percent	output	This output network variable reflects the current level of the primary mechanical cooling output or can be used to control a remote mechanical cooling source.
34 "Secondary Cool Output" (nvoCoolSecondary)	Optional	SNVT_lev_percent	output	This output network variable reflects the current level of the secondary mechanical cooling output or can be used to control a remote mechanical cooling source.
35 "Space Humidity Output" (nvoSpaceRH)	Optional	SNVT_lev_percent	output	This output network variable indicates the space humidity in percent, if the Chilled Ceiling Controller Device has a locally wired humidity sensor.

NV Member	Requirement	CNP Type	Direction	Description
36 "Space Dewpoint Temperature Output" (nvoSpaceDewPt)	Optional	SNVT_temp_p	output	This output network variable indicates the space dew point temperature. This value can be measured or calculated by the Chilled Ceiling Controller.
37 "Energy Hold Off Output" (nvoEnergyHoldOff)	Optional	SNVT_switch	output	This output indicates the state of an Energy Hold Off device that is hardwired to the controller.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to nvoEffectSetpt	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoHeatCool	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoSetptShift	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoTerminalLoad	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoHeatPrimary	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoHeatSecondary	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoCoolPrimary	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoCoolSecondary	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoSpaceRH	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to nvoSpaceDewPt	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoEnergyHoldOff	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxRcvTime applying to nviSpaceTemp	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviSetptOffset	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviSetptShift	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviOccSchedule	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviOccSensor	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviApplicMode	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviHeatCool	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxRcvTime applying to nviEnergyHoldOff	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviSourceTemp	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviHeatSrcTemp	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviCoolSrcTemp	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviSpaceRH	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviSpaceDewPt	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviOutdoorDewPt	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviHeatPriSlave	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxRcvTime applying to nviHeatSecSlave	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviCoolPriSlave	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviCoolSecSlave	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTsetPnts applying to SFPTchilledCeiling Controller	Mandatory	none	not permitted	The occupancy temperature setpoints for heat and cool mode
SCPTminSendTime applying to SFPTchilledCeiling Controller	Optional	none	not permitted	The minimum period of time between consecutive transmissions of the current value
SCPTlocation applying to SFPTchilledCeiling Controller	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTbypassTime applying to SFPTchilledCeiling Controller	Optional	none	not permitted	The maximum amount of time that the controller can be in the bypass (occupancy) mode following the last bypass request. Zero disables the timer.
SCPTmanOvrTime applying to SFPTchilledCeiling Controller	Optional	none	not permitted	The maximum time that the controller will stay in a manual mode following the last request by a network variable input. Zero disables the timer.
SCPTmaxSendTime applying to SFPTchilledCeiling Controller	Mandatory	none	not permitted	The maximum period of time between consecutive transmissions of the current value

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to nvoUnitStatus	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoSpaceTemp	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxRcvTime applying to SFPTchilledCeilingController	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables

7.60 SFPTunitVentilatorController (8080)

NV Member	Requirement	CNP Type	Direction	Description
1 "Space Temperature Input" (nviSpaceTemp)	Mandatory	SNVT_temp_p	input	This input network variable is used to connect an external space temperature sensor to the node.
2 "Temperature Setpoint Input (absolute)" (nviSetpoint)	Optional	SNVT_temp_p	input	This input network variable is used to allow the temperature setpoints for the occupied and standby modes to be changed via the network.
3 "Setpoint Offset Input" (nviSetptOffset)	Optional	SNVT_temp_p	input	This input network variable is used to shift the effective occupied and standby temperature setpoints by adding nviSetptOffset to the current setpoints.
4 "Setpoint Shift Input" (nviSetptShift)	Optional	SNVT_temp_setpt	input	This input network variable is used to shift the effective heat/cool setpoints by adding the corresponding value in nviSetptShift to the current setpoints.
5 "Occupancy Scheduler Input" (nviOccSchedule)	Optional	SNVT_tod_event	input	This input network variable is used to command the Unit Ventilator Controller into different occupancy modes.

NV Member	Requirement	CNP Type	Direction	Description
6 "Occupancy Override Input" (nviOccManCmd)	Optional	SNVT_occupancy	input	This input network variable is used to command the Unit Ventilator Controller into different occupancy modes.
7 "Occupancy Sensor Input" (nviOccSensor)	Optional	SNVT_occupancy	input	This input network variable is used to indicate the presence of occupants in the controlled space.
8 "Application Mode Input" (nviApplicMode)	Optional	SNVT_hvac_mode	input	This network variable input is used to coordinate the Unit Ventilator Controller with any supervisory controller.
9 "Heat/Cool Mode Input" (nviHeatCool)	Optional	SNVT_hvac_mode	input	This network variable input is used to coordinate the Unit Ventilator Controller with any node that may need to control the heat/cool changeover of the unit.
10 "Fan Speed Command Input" (nviFanSpeedCmd)	Optional	SNVT_switch	input	This input network variable is used to connect an external fan speed switch to the node or to allow any supervisory device to override the fan speed controlled by the node's control algorithm.
11 "Compressor Enable Input" (nviComprEnable)	Optional	SNVT_switch	input	This input is used to disable compressor operation. This input is typically sent from a system coordination panel.
12 "Auxiliary Heat Enable Input" (nviAuxHeatEnable)	Optional	SNVT_switch	input	This input is used to disable auxiliary heat operation. This input is typically sent from a system supervisor panel.
13 "Economizer Enable Input" (nviEconEnable)	Optional	SNVT_switch	input	This input is used to enable and disable economizer operation.
14 "Energy Hold Off Input" (nviEnergyHoldOff)	Optional	SNVT_switch	input	This input is used to stop heating and cooling while allowing the unit to protect the space from temperature extremes.

NV Member	Requirement	CNP Type	Direction	Description
15 "Water Valve Override Input" (nviValveOverride)	Optional	SNVT_hvac_overid	input	This input network variable is used for commanding the controller into a manual mode for overriding water valves controlled by the unit.
16 "Emergency Override Input" (nviEmergOverride)	Optional	SNVT_hvac_emerg	input	This input network variable is used to command the device into different emergency modes. It is typically set by a supervisory node.
17 "Source Temperature Input" (nviSourceTemp)	Optional	SNVT_temp_p	input	This input network variable is used to indicate the temperature of the air or water being supplied to the unit for heating and/or cooling capacity.
18 "Outdoor Air Temperature Input" (nviOutdoorTemp)	Optional	SNVT_temp_p	input	This input network variable represents information from an outdoor air temperature sensor.
19 "Percentage level" (nviSpaceRH)	Optional	SNVT_lev_percent	input	This input network variable is the measured space humidity in percent. This input is typically sent from a communicating humidity sensor.
20 "Outdoor Air Humidity Input" (nviOutdoorRH)	Optional	SNVT_lev_percent	input	This input network variable is the measured outdoor humidity in percent.
21 "Effective Space Temperature Output" (nvoSpaceTemp)	Mandatory	SNVT_temp_p	output	This output network variable is used to monitor the effective space temperature that the Chilled Ceiling Controller is using for control.
22 "Effective Setpoint Output" (nvoEffectSetpt)	Mandatory	SNVT_temp_p	output	This output network variable is used to monitor the effective temperature setpoint.
23 "Unit Status Output" (nvoUnitStatus)	Mandatory	SNVT_hvac_status	output	This output network variable is available to report the Unit Ventilator Controller status.
24 "Effective Occupancy Output" (nvoEffectOccup)	Optional	SNVT_occupancy	output	This output network variable is used to indicate the actual occupancy mode of the unit.

NV Member	Requirement	CNP Type	Direction	Description
25 "Effective Heat/Cool Output" (nvoHeatCool)	Optional	SNVT_hvac_mode	output	This output network variable is used to indicate the actual heat/cool mode of the unit.
26 "Local Setpoint Output" (nvoSetpoint)	Optional	SNVT_temp_p	output	This output network variable is used to monitor the space temperature setpoint if a setpoint device is locally wired.
27 "Local Setpoint Shift Output" (nvoSetptShift)	Optional	SNVT_temp_setpt	output	This output network variable is used to report a locally determined shift of the effective heat/cool setpoints.
28 "Fan Speed Output" (nvoFanSpeed)	Optional	SNVT_switch	output	This output network variable reflects the actual fan speed of a local multi-speed fan as well as the requested speed of a remote fan.
29 "Discharge Air Temperature Output" (nvoDischAirTemp)	Optional	SNVT_temp_p	output	This output network variable is used to monitor the temperature of the air that leaves the Unit Ventilator Controller, if the unit controller provides a hardwired temperature sensor for this purpose.
30 "Absolute Power Consumption KW Output" (nvoLoadAbsK)	Optional	SNVT_power_kilo	output	This output network variable can used to indicate the current power consumption of the unit.
31 "Terminal Load Output" (nvoTerminalLoad)	Optional	SNVT_lev_percent	output	This output indicates the current heat/cool energy demand of the unit.
32 "Primary Heat Output" (nvoHeatPrimary)	Optional	SNVT_lev_percent	output	This output network variable reflects the current level of the primary heat output or can be used to control a remote primary heat source.
33 "Secondary Heat Output" (nvoHeatSecondary)	Optional	SNVT_lev_percent	output	This output network variable reflects the current level of the secondary heat output or can be used to control a remote secondary heat source.

NV Member	Requirement	CNP Type	Direction	Description
34 "Primary Cool Output" (nvoCoolPrimary)	Optional	SNVT_lev_percent	output	This output network variable reflects the current level of the primary mechanical cooling output or can be used to control a remote mechanical cooling source.
35 "Outdoor Air Damper Output" (nvoOADamper)	Optional	SNVT_lev_percent	output	This output network variable reflects the current position of the outdoor air damper or as a request to a remote outdoor air damper.
36 "Space Humidity Output" (nvoSpaceRH)	Optional	SNVT_lev_percent	output	This output network variable indicates the space humidity in percent, if the Chilled Ceiling Controller Device has a locally wired humidity sensor.
37 "Outdoor Air Humidity Output" (nvoOutdoorRH)	Optional	SNVT_lev_percent	output	This output network variable indicates the outdoor air humidity in percent, if the Unit Ventilator Controller Device has a locally wired humidity sensor.
38 "Outdoor Air Temperature Output" (nvoOutdoorTemp)	Optional	SNVT_temp_p	output	This output network variable is used to monitor the outdoor air temperature if the unit controller provides a hardwired temperature sensor for this purpose.
39 "Space CO2 Sensor Output" (nvoSpaceCO2)	Optional	SNVT_ppm	output	This output network variable indicates the space CO2 concentration in ppm, if the Unit Ventilator Controller Device has a locally wired CO2 sensor.
40 "Energy Hold Off Output" (nvoEnergyHoldOff)	Optional	SNVT_switch	output	This output indicates the state of an Energy Hold Off device that is hardwired to the controller.
41 "Space CO2 Sensor Input" (nviSpaceCO2)	Optional	SNVT_ppm	input	This input network variable measures the space CO2 levels in PPM.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to nvoHeatCool	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to nvoSetptShift	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoFanSpeed	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoTerminalLoad	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoHeatPrimary	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoHeatSecondary	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoCoolPrimary	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoOADamper	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoSpaceRH	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoOutdoorRH	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoOutdoorTemp	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoSpaceCO2	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoEnergyHoldOff	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTsetPnts applying to SFPTunitVentilatorController	Mandatory	none	not permitted	The occupancy temperature setpoints for heat and cool mode
SCPTmaxRcvTime applying to nviSpaceTemp	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviSetptOffset	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviSetptShift	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviOccSchedule	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviOccSensor	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviApplicMode	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviHeatCool	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxRcvTime applying to nviComprEnable	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviAuxHeatEnable	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviEconEnable	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviEnergyHoldOff	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviSourceTemp	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviOutdoorTemp	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviSpaceRH	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviOutdoorRH	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTminSendTime applying to SFPTunitVentilatorController	Optional	none	not permitted	The minimum period of time between consecutive transmissions of the current value
SCPTlocation applying to SFPTunitVentilatorController	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTbypassTime applying to SFPTunitVentilatorController	Optional	none	not permitted	The maximum amount of time that the controller can be in the bypass (occupancy) mode following the last bypass request. Zero disables the timer.
SCPTmanOvrTime applying to SFPTunitVentilatorController	Optional	none	not permitted	The maximum time that the controller will stay in a manual mode following the last request by a network variable input. Zero disables the timer.
SCPTminRnge applying to SFPTunitVentilatorController	Optional	none	not permitted	The minimum limit of the value of the primary output network variable for the object
SCPTlimitCO2 applying to SFPTunitVentilatorController	Optional	none	not permitted	CO2 threshold limit, controller to maintain concentration below this limit
SCPThumSetpt applying to SFPTunitVentilatorController	Optional	none	not permitted	High limit humidity setpoint for the controlled space. A zero value disables
SCPTnumValves applying to SFPTunitVentilatorController	Optional	none	not permitted	Used to inform the controller whether it is in a one-valve or two-valve system
SCPTmaxSendTime applying to SFPTunitVentilatorController	Mandatory	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoSpaceTemp	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to nvoEffectSetpt	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoUnitStatus	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxRcvTime applying to SFPTunitVentilatorController	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviSpaceCO2	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables

7.61 SFPTsccCommandModule (8090)

NV Member	Requirement	CNP Type	Direction	Description
1 "Space Temperature Input" (nviSpaceTemp)	Optional	SNVT_temp_p	input	This input network variable is used to display the value of an external space temperature sensor.
2 "User Lockout Input" (nviUserLockout)	Optional	SNVT_switch	input	This input network variable is used by the supervisory device to restrict the occupant from making certain changes.
3 "Time Input" (nviTime)	Optional	SNVT_time_stamp	input	This input network variable is used to display the local time in the Command Module Device
4 "Effective Temperature Setpoint Input" (nviEffectSetpt)	Optional	SNVT_temp_p	input	This input network variable is from a controller object and is used to display in the Command Module Device the controller's Effective Setpoint.

NV Member	Requirement	CNP Type	Direction	Description
5 "Effective Occupancy Input" (nviEffectOccup)	Optional	SNVT_occupancy	input	This input network variable is used to indicate the associated controller object's actual occupancy mode.
6 "Unit Status Input" (nviUnitStatus)	Optional	SNVT_hvac_status	input	This input network variable is transmitted from the Controller object to inform the occupant of the status of the associated HVAC equipment, to be displayed in the Command Module Device.
7 "Outdoor Air Temperature Input" (nviOutdoorTemp)	Optional	SNVT_temp_p	input	This input network variable represents information from an outdoor air temperature sensor.
8 "Outdoor Air Humidity Input" (nviOutdoorRH)	Optional	SNVT_lev_percent	input	This input network variable is used to display outdoor humidity in percent.
9 "Space Humidity Input" (nviSpaceRH)	Optional	SNVT_lev_percent	input	This input network variable is used to display space humidity in percent.
10 "Space CO2 Sensor Input" (nviSpaceCO2)	Optional	SNVT_ppm	input	This input network variable is used to display the space CO2 levels in PPM.
11 "Energy Hold Off Input" (nviEnergyHoldOff)	Optional	SNVT_switch	input	This input is from a space comfort controller device which monitors inputs such as a window contact sensor.
12 "Temperature Setpoint Output" (nvoSetpoint)	Mandatory	SNVT_temp_p	output	This output network variable is used to allow the occupant to change the temperature setpoint for the occupied and standby mode from the Command Module Device
13 "Space Temperature Output" (nvoSpaceTemp)	Mandatory	SNVT_temp_p	output	This output network variable is used to transmit the space temperature that is hard wired to the Command Module Device.

NV Member	Requirement	CNP Type	Direction	Description
14 "Heat/Cool Output" (nvoHeatCool)	Optional	SNVT_hvac_mode	output	This output network variable from the Command Module Device is used to transmit the user's command to the controller of the HVAC equipment.
15 "Fan Speed Output" (nvoFanSpeed)	Optional	SNVT_switch	output	This output network variable reflects the requested speed of a remote fan.
16 "Occupancy Sensor Output" (nvoOccSensor)	Optional	SNVT_occupancy	output	The Command Module Device object conveys to the network the occupancy state of a hard wired occupancy sensor.
17 "Space Humidity Output" (nvoSpaceRH)	Optional	SNVT_lev_percent	output	This output network variable is used to transmit the space relative humidity that is hard wired to the Command Module Device.
18 "Space CO2 Sensor Output" (nvoSpaceCO2)	Optional	SNVT_ppm	output	This output network variable is used to transmit the space CO2 sensor value that is hard wired to the Command Module Device.
19 "Occupancy Manual Command Output" (nvoOccManCmd)	Optional	SNVT_occupancy	output	The Command Module Device object conveys to the network the occupancy state as it is modified by the occupant.
20 "Temperature Setpoint Offset Output" (nvoSetptOffset)	Optional	SNVT_temp_p	output	This output network variable is used to shift the temperature setpoint via the network, by adding nviSetPtOffset to the current setpoint.
21 "Occupancy Temperature Setpoints Output" (nvoSetpoints)	Optional	SNVT_temp_setpt	output	This output defines the occupancy temperature setpoints for heat and cool mode.
22 "Space CO2 Limit Output" (nvoSpaceCO2Lim)	Optional	SNVT_ppm	output	This network output property defines a high limit CO2 setpoint as requested by the occupant for ventilation functions.
23 "Space Humidity Setpoint Output" (nvoSpaceRHSetpt)	Optional	SNVT_lev_percent	output	This network output property defines the high limit humidity setpoint for the controlled space.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to SFPTsccCommandModule	Mandatory	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoHeatCool	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoOccSensor	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoSpaceRH	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoSpaceCO2	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoSetptOffset	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPToffsetTemp applying to SFPTsccCommandModule	Mandatory	none	not permitted	Used to calibrate external hardware with additive offset after transformation
SCPTminDeltaTemp applying to SFPTsccCommandModule	Mandatory	none	not permitted	The minimum change in temperature required to be treated as significant
SCPTminSendTime applying to SFPTsccCommandModule	Optional	none	not permitted	The minimum period of time between consecutive transmissions of the current value
SCPTsetPnts applying to SFPTsccCommandModule	Optional	none	not permitted	The occupancy temperature setpoints for heat and cool mode

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxRcvTime applying to nviSpaceTemp	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviTime	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviEffectSetpt	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviUnitStatus	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviOutdoorTemp	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviOutdoorRH	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviSpaceRH	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviSpaceCO2	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxRcvTime applying to nviEnergyHoldOff	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTlocation applying to SFPTsccCommandModule	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPToffsetCO2 applying to SFPTsccCommandModule	Optional	none	not permitted	Used to calibrate external hardware with additive offset after transformation
SCPTminDeltaCO2 applying to SFPTsccCommandModule	Optional	none	not permitted	The minimum change in CO2 level required to be treated as significant
SCPToffsetRH applying to SFPTsccCommandModule	Optional	none	not permitted	Used to calibrate external hardware with additive offset after transformation
SCPTminDeltaRH applying to SFPTsccCommandModule	Optional	none	not permitted	The minimum change in RH level required to be treated as significant
SCPTmaxSendTime applying to nvoSpaceTemp	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxRcvTime applying to SFPTsccCommandModule	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables

7.62 SFPTdamperActuator (8110)

NV Member	Requirement	CNP Type	Direction	Description
1 "Actuator setpoint input" (nviRelStpt)	Mandatory	SNVT_lev_percent	input	Controls the relative actuator setpoint
2 "Actuator state input" (nviActuatState)	Mandatory	SNVT_switch	input	Used where the actuator is connected to a switch that forces the actuator to a predefined set of positions or air volumes
3 "Actual position output" (nvoActualValue)	Mandatory	SNVT_lev_percent	output	Present position of the actuator
4 "Emergency override input" (nviEmergOvrd)	Optional	SNVT_hvac_emerg	input	Used in fire and smoke applications to open and close an actuator with maximum speed; this input has highest priority
5 "Manual override input" (nviManOvrd)	Optional	SNVT_hvac_overid	input	Command the actuator into a manual mode (mainly used during balancing)
6 "Actuator state feedback output" (nvoActuatStateFb)	Optional	SNVT_switch	output	Mirrors the actual value of nviActuatState
7 "Actuator setpoint feedback output" (nvoRelStptFb)	Optional	SNVT_lev_percent	output	Mirrors the actual value of nviRelStpt
8 "Damper angle output" (nvoAbsAngle)	Optional	SNVT_angle_deg	output	Present position of the actuator's shaft or damper blade
9 "Air flow output" (nvoAbsAirFlow)	Optional	SNVT_flow	output	Airflow through the associated VAV box
10 "Inside duct temperature output" (nvoDuctTemp)	Optional	SNVT_temp_p	output	Present temperature inside the duct

NV Member	Requirement	CNP Type	Direction	Description
11 "Emergency override feedback output" (nvoEmergOvrdFb)	Optional	SNVT_hvac_emerg	output	Mirrors the value of nviEmergOvrd

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTminSendTime applying to SFPTdamperActuator	Mandatory	none	not permitted	The minimum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to SFPTdamperActuator	Mandatory	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxRcvTime applying to SFPTdamperActuator	Mandatory	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTdirection applying to SFPTdamperActuator	Mandatory	none	not permitted	The actuator sense of rotation and safety position; bit 0 set => counterclockwise, bit 1 set => damper open
SCPTminDeltaFlow applying to nvoAbsAirFlow	Optional	none	not permitted	The minimum change in airflow required to be treated as significant
SCPTminDeltaTemp applying to nvoDuctTemp	Optional	none	not permitted	The minimum change in temperature required to be treated as significant
SCPTminDeltaAngl applying to nvoAbsAngle	Optional	none	not permitted	The minimum change in damper actuator angle required to be treated as significant
SCPTactuatorType applying to SFPTdamperActuator	Optional	manufacturer specific	not permitted	The identification of the exact actuator type or label

CNP Type	Requirement	Restrictions	Arrays	Description
SCPToemType applying to SFPTdamperActuator	Optional	device specific, constant	not permitted	The label, programmed by the OEM, to identify the unit name
SCPTlocation applying to SFPTdamperActuator	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTdriveTime applying to SFPTdamperActuator	Optional	none	not permitted	The transition time for a full 100 % stroke (change from one extreme to the other)
SCPTnomAngle applying to SFPTdamperActuator	Optional	none	not permitted	The nominal angle for an actuator
SCPTnomAirFlow applying to SFPTdamperActuator	Optional	none	not permitted	Value used in calculating the air flow in an airflow control actuator
SCPTductArea applying to SFPTdamperActuator	Optional	none	not permitted	The duct area used to calculate the air flow, relevant only for VAV actuators / controllers
SCPTmaxRnge applying to nvoActualValue	Optional	none	not permitted	The maximum limit of the value of the primary output network variable for the object
SCPTminRnge applying to nvoActualValue	Optional	none	not permitted	The minimum limit of the value of the primary output network variable for the object
SCPTsensConstVAV applying to SFPTdamperActuator	Optional	none	not permitted	Calibration constant used to calculate airflow
SCPTsensConstTmp applying to SFPTdamperActuator	Optional	none	not permitted	Calibration value for a duct temperature sensor

7.63 SFPTpumpController (8120)

NV Member	Requirement	CNP Type	Direction	Description
1 "Pump setpoint for normal operation" (nviPumpSetpoint)	Mandatory	SNVT_switch	input	
2 "Requested pump operating mode" (nviPumpOpMode)	Mandatory	SNVT_hvac_mode	input	
3 "Pump capacity in percent" (nvoPumpCapacity)	Mandatory	SNVT_lev_percent	output	
4 "Effective operating mode" (nvoEffOpMode)	Mandatory	SNVT_hvac_mode	output	
5 "Effective device control mode" (nvoControlMode)	Mandatory	SNVT_dev_c_mode	output	
6 "Pump override stop command" (nviPumpOvdStop)	Optional	SNVT_switch	input	
7 "Override speed in percent" (nviOvdSpeed)	Optional	SNVT_lev_percent	input	
8 "Override setpoint for pressure" (nviOvdPress)	Optional	SNVT_press	input	
9 "Override setpoint for flow" (nviOvdFlow)	Optional	SNVT_flow_p	input	

NV Member	Requirement	CNP Type	Direction	Description
10 "Remote differential pressure sensor" (nviRemotePress)	Optional	SNVT_press	input	
11 "Remote flow sensor" (nviRemoteFlow)	Optional	SNVT_flow_p	input	
12 "Remote temperature sensor" (nviRemoteTemp)	Optional	SNVT_temp_p	input	
13 "Device status" (nvoPumpStatus)	Optional	SNVT_dev_status	output	Current status of the device
14 "Pump pressure" (nvoPressure)	Optional	SNVT_press	output	
15 "Pump flow" (nvoFlow)	Optional	SNVT_flow_p	output	
16 "Pump speed" (nvoSpeed)	Optional	SNVT_rpm	output	
17 "Pump override active" (nvoPumpOverride)	Optional	SNVT_switch	output	
18 "Runtime" (nvoRuntime)	Optional	SNVT_time_hour	output	Total running time for the pump in hours. After 65 535 h, the counter starts again at zero.
19 "Device fault states" (nvoPumpFault)	Optional	SNVT_dev_fault	output	Fault information for the device
20 "Device maintenance" (nvoMaintenance)	Optional	SNVT_dev_maint	output	Device-maintenance states

NV Member	Requirement	CNP Type	Direction	Description
21 "Fluid temperature" (nvoFluidTemp)	Optional	SNVT_temp_p	output	
22 "Electrical power consumption in Watts" (nvoPower)	Optional	SNVT_power	output	
23 "Electrical power consumption in kiloWatts" (nvoPowerkilo)	Optional	SNVT_power_kilo	output	
24 "Total energy consumption of the pump" (nvoEnergyConsum)	Optional	SNVT_elec_kwh	output	

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to SFPTpumpController	Mandatory	none	not permitted	
SCPTpumpCharacteristic applying to SFPTpumpController	Mandatory	manufacturer specific	not permitted	
SCPTlocation applying to SFPTpumpController	Optional	none	not permitted	
SCPTmaxRcvTime applying to SFPTpumpController	Optional	none	not permitted	
SCPTminSendTime applying to SFPTpumpController	Optional	none	not permitted	

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTminPressure Setpoint applying to SFPTpumpControll er	Optional	none	not permitted	
SCPTmaxPressure Setpoint applying to SFPTpumpControll er	Optional	none	not permitted	
SCPTminFlowSetp oint applying to SFPTpumpControll er	Optional	none	not permitted	
SCPTmaxFlowSet point applying to SFPTpumpControll er	Optional	none	not permitted	
SCPTdeviceContro lMode applying to SFPTpumpControll er	Optional	none	not permitted	
SCPTminRemoteP ressureSetpoint applying to nviRemotePress	Optional	none	not permitted	
SCPTmaxRemote PressureSetpoint applying to nviRemotePress	Optional	none	not permitted	
SCPTminRemoteF lowSetpoint applying to nviRemoteFlow	Optional	none	not permitted	
SCPTmaxRemote FlowSetpoint applying to nviRemoteFlow	Optional	none	not permitted	
SCPTminRemoteT empSetpoint applying to nviRemoteTemp	Optional	none	not permitted	

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxRemoteTempSetpoint applying to nviRemoteTemp	Optional	none	not permitted	
SCPTobjMajVer applying to SFPTpumpController	Optional	constant	not permitted	The major version number for the object
SCPTobjMinVer applying to SFPTpumpController	Optional	device specific, constant	not permitted	The minor version number for the object

7.64 SFPThvacValvePositioner (8131)

NV Member	Requirement	CNP Type	Direction	Description
1 "Valve Control Signal" (nviControlSignal)	Mandatory	SNVT_lev_percent	input	This input network variable provides a control setpoint. The setpoint is given as a percentage of the total required flow or heating/cooling energy.
2 "Desired HVAC Operating Mode" (nviHvacOpMode)	Mandatory	SNVT_hvac_mode	input	A supervisory controller typically uses this input network variable to set the valve operating mode.
3 "Valve-Control Signal Mirror" (nvoControlSignFb)	Mandatory	SNVT_lev_percent	output	This output network variable mirrors the actual value of nviControlSignal. This can be used for multiple valve-positioner applications.
4 "Effective HVAC Operating Mode" (nvoHvacOpMode)	Mandatory	SNVT_hvac_mode	output	This output network variable provides the actual HVAC operating mode.
5 "Effective Valve Operating Mode" (nvoValveOpMode)	Mandatory	SNVT_valve_mode	output	This output network variable provides the actual valve operating mode.

NV Member	Requirement	CNP Type	Direction	Description
6 "Override-Valve Stop" (nviOvrdStop)	Optional	SNVT_switch	input	This input network variable provides a manual override function to stop the valve; typically from a supervisory device.
7 "Override Valve Capacity" (nviOvrdCapacity)	Optional	SNVT_lev_percent	input	This input network variable provides an override request to the valve capacity; typically from a supervisory device.
8 "Override Valve Position" (nviOvrdPosition)	Optional	SNVT_lev_percent	input	This input network variable provides for both an override request and a position value relative to the maximum stroke; typically from a supervisory device.
9 "Emergency Mode Request" (nviEmergencyMode)	Optional	SNVT_hvac_emerg	input	Mandatory for 01 /21: This input network variable is used when the positioner possesses a real emergency operation (for example: a spring-return drive).
10 "Read-Value Valve Position" (nviValvePosition)	Optional	SNVT_lev_percent	input	This input network variable is used to connect an external position sensor for precise position control.
11 "Valve Capacity" (nvoValveCapacity)	Optional	SNVT_lev_percent	output	This output network variable provides the actual valve capacity as a percentage of the flow.
12 "Valve Position" (nvoValvePosition)	Optional	SNVT_lev_percent	output	This output network variable provides the calculated position relative to the maximum stroke length.
13 "Stroke Length" (nvoStrokeLength)	Optional	SNVT_length_mil	output	This output network variable provides the calculated stroke as length in millimetres.
14 "Actual Valve Position" (nvoActValvePos)	Optional	SNVT_lev_percent	output	This output network variable provides the actual position relative to the maximum stroke length.
15 "Valve Flow" (nvoValveFlow)	Optional	SNVT_lev_percent	output	This output network variable provides the calculated flow relative to the maximum flow of the valve.

NV Member	Requirement	CNP Type	Direction	Description
16 "Drive Status" (nvoDriveStatus)	Optional	SNVT_dev_status	output	This output network variable provides status/diagnostic information of the valve drive.
17 "Valve Fault" (nvoValveFault)	Optional	SNVT_dev_fault	output	This output network variable provides fault information about the valve.
18 "Device maintenance" (nvoMaintenance)	Optional	SNVT_dev_maint	output	This output network variable provides maintenance information of the valve.
19 "Runtime" (nvoRuntime)	Optional	SNVT_elapsed_tm	output	This output network variable provides the total cumulative running time of the valve actuator.
20 "Cumulative Stroke Distance" (nvoStrokeCumul)	Optional	SNVT_length	output	This output network variable provides the cumulative distance of all strokes of the valve as length in meters.
21 "Position-Error Count" (nvoPosErrCount)	Optional	SNVT_count	output	This output network variable provides the count of positioning errors.
22 "Power-Failure Count" (nvoPwrFailCount)	Optional	SNVT_count	output	This output network variable provides the count of power failures and voltage dips
23 "Emergency Count" (nvoEmergCount)	Optional	SNVT_count	output	This output network variable provides the count of emergency actions, as represented with the Emergency Mode State network variable
24 "Emergency Mode State" (nvoEmergencyMode)	Optional	SNVT_hvac_emerg	output	Mandatory for 01 /21: This output network variable is only used in combination with the Emergency Mode Request input network variable

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to nvoControlSignFb	Mandatory	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoHvacOpMode	Mandatory	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoValveOpMode	Mandatory	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoValveCapacity	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoEmergencyMode	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxRcvTime applying to nviControlSignal	Mandatory	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviHvacOpMode	Mandatory	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTmaxRcvTime applying to nviEmergencyMode	Optional	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTcontrolSignal applying to SFPThvacValvePositioner	Mandatory	none	not permitted	Start and end points (X,Y) for a transition
SCPTdefOutput applying to SFPThvacValvePositioner	Mandatory	none	not permitted	The position or level the sensor should adopt when updates are not received, or at power-on reset, or when overridden

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTminSendTime applying to nvoControlSignFb	Mandatory	none	not permitted	The minimum period of time between consecutive transmissions of the current value
SCPTminSendTime applying to nvoHvacOpMode	Mandatory	none	not permitted	The minimum period of time between consecutive transmissions of the current value
SCPTminSendTime applying to nvoValveOpMode	Mandatory	none	not permitted	The minimum period of time between consecutive transmissions of the current value
SCPTminSendTime applying to nvoValveCapacity	Optional	none	not permitted	The minimum period of time between consecutive transmissions of the current value
SCPTminSendTime applying to nvoValvePosition	Optional	none	not permitted	The minimum period of time between consecutive transmissions of the current value
SCPTminSendTime applying to nvoStrokeLength	Optional	none	not permitted	The minimum period of time between consecutive transmissions of the current value
SCPTminSendTime applying to nvoActValvePos	Optional	none	not permitted	The minimum period of time between consecutive transmissions of the current value
SCPTminSendTime applying to nvoValveFlow	Optional	none	not permitted	The minimum period of time between consecutive transmissions of the current value
SCPTminSendTime applying to nvoDriveStatus	Optional	none	not permitted	The minimum period of time between consecutive transmissions of the current value
SCPTminSendTime applying to nvoValveFault	Optional	none	not permitted	The minimum period of time between consecutive transmissions of the current value
SCPTminSendTime applying to nvoMaintenance	Optional	none	not permitted	The minimum period of time between consecutive transmissions of the current value
SCPTminSendTime applying to nvoRuntime	Optional	none	not permitted	The minimum period of time between consecutive transmissions of the current value

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTminSendTime applying to nvoStrokeCumul	Optional	none	not permitted	The minimum period of time between consecutive transmissions of the current value
SCPTminSendTime applying to nvoPosErrCount	Optional	none	not permitted	The minimum period of time between consecutive transmissions of the current value
SCPTminSendTime applying to nvoPwrFailCount	Optional	none	not permitted	The minimum period of time between consecutive transmissions of the current value
SCPTminSendTime applying to nvoEmergCount	Optional	none	not permitted	The minimum period of time between consecutive transmissions of the current value
SCPTminSendTime applying to nvoEmergencyMode	Optional	none	not permitted	The minimum period of time between consecutive transmissions of the current value
SCPTlocation applying to SFPThvacValvePositioner	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTminSetpoint applying to SFPThvacValvePositioner	Optional	device specific, constant	not permitted	The minimum setpoint, such as minimum angle of rotation or minimum air flow
SCPTmaxSetpoint applying to SFPThvacValvePositioner	Optional	device specific, constant	not permitted	Either the maximum angle of rotation for an actuator or the maximum airflow for an actuator depending on actuator category
SCPTminStroke applying to SFPThvacValvePositioner	Optional	device specific, constant	not permitted	The minimum stroke limit
SCPTmaxStroke applying to SFPThvacValvePositioner	Optional	device specific, constant	not permitted	The maximum stroke limit

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTvalveOperatingMode applying to SFPThvacValvePositioner	Optional	none	not permitted	The normal operating mode of the valve
SCPTnightPurgePosition applying to SFPThvacValvePositioner	Optional	none	not permitted	Valve position in percent open for night purge
SCPTfreeCoolPosition applying to SFPThvacValvePositioner	Optional	none	not permitted	Valve position in percent open for free cooling HVAC mode
SCPTemergencyPosition applying to SFPThvacValvePositioner	Optional	none	not permitted	Position in percent of full scale (open) for emergency operation
SCPTdriveTime applying to SFPThvacValvePositioner	Optional	device specific, constant	not permitted	The transition time for a full 100% stroke (change from one extreme to the other)
SCPTvalveStroke applying to SFPThvacValvePositioner	Optional	device specific, constant	not permitted	
SCPTvalveNominalSize applying to SFPThvacValvePositioner	Optional	none	not permitted	
SCPTvalveKvs applying to SFPThvacValvePositioner	Optional	none	not permitted	
SCPTvalveType applying to SFPThvacValvePositioner	Optional	none	not permitted	
SCPTmanfDate applying to SFPThvacValvePositioner	Optional	none	not permitted	The date of manufacture for the device

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTinstallDate applying to SFPThvacValvePositioner	Optional	none	not permitted	The date of installation for the device
SCPTactuatorType applying to SFPThvacValvePositioner	Optional	device specific, constant	not permitted	The identification of the exact actuator type or label
SCPTactuatorCharacteristic applying to SFPThvacValvePositioner	Optional	device specific, constant	not permitted	
SCPTtrnsTblX applying to SFPThvacValvePositioner	Optional	none	not permitted	Used in conjunction with Translation table Y to scale and linearize a value
SCPTtrnsTblY applying to SFPThvacValvePositioner	Optional	none	not permitted	Used in conjunction with Translation table X to scale and linearize a value
SCPTvalveFlowCharacteristic applying to SFPThvacValvePositioner	Optional	device specific, constant	not permitted	Actual flow characteristic of the valve
SCPTtrnsTblX2 applying to SFPThvacValvePositioner	Optional	none	not permitted	
SCPTtrnsTblY2 applying to SFPThvacValvePositioner	Optional	none	not permitted	
SCPTcombFlowCharacteristic applying to SFPThvacValvePositioner	Optional	none	not permitted	

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTtrnsTblX3 applying to SFPThvacValvePositioner	Optional	none	not permitted	
SCPTtrnsTblY3 applying to SFPThvacValvePositioner	Optional	none	not permitted	
SCPTblockProtectionTime applying to SFPThvacValvePositioner	Optional	none	not permitted	The minimum time in hours for movement to prevent blocking
SCPTrunTimeAlarm applying to SFPThvacValvePositioner	Optional	none	not permitted	
SCPTobjMajVer applying to SFPThvacValvePositioner	Optional	constant	not permitted	The major version number for the object
SCPTobjMinVer applying to SFPThvacValvePositioner	Optional	device specific, constant	not permitted	The minor version number for the object

7.65 SFPTboilerController (8301)

NV Member	Requirement	CNP Type	Direction	Description
1 "Boiler enable input" (nviBoilerEnable)	Mandatory	SNVT_switch	input	Used to disable (stop) boiler operation, or to enable (automatic, local, or remote) boiler operation
2 "Boiler state output" (nvoBoilerState)	Mandatory	SNVT_switch	output	Boiler's present level of heat output, as well as the requested firing rate of a remote boiler
3 "Effective setpoint output" (nvoEffectSetpt)	Mandatory	SNVT_temp_p	output	Used to monitor the effective temperature setpoint

NV Member	Requirement	CNP Type	Direction	Description
4 "Application mode input" (nviApplicMode)	Optional	SNVT_hvac_mode	input	Used to coordinate the boiler controller node with any supervisory controller
5 "Pump speed command input" (nviPumpSpeedCmd)	Optional	SNVT_switch	input	Used to connect an external pump control to the node or to allow an override of the pump speed
6 "Supply temperature input" (nviSupplyTemp)	Optional	SNVT_temp_p	input	Used to connect an external supply temperature sensor to the node
7 "Outside air temperature input" (nviOutdoorTemp)	Optional	SNVT_temp_p	input	Represents information from an outdoor air temperature sensor
8 "Return temperature input" (nviReturnTemp)	Optional	SNVT_temp_p	input	Used to connect an external return temperature sensor to the node
9 "Temperature setpoint input (absolute)" (nviSetpoint)	Optional	SNVT_temp_p	input	Used to allow the heating setpoint for the boiler water temperature to be changed via the network
10 "Boiler command input" (nviBoilerCmd)	Optional	SNVT_switch	input	Used to command the boiler state and firing rate of the boiler controller (e.g., to disable the boiler, or allow automatic control)
11 "Boiler load output" (nvoBoilerLoad)	Optional	SNVT_lev_percent	output	Present heat/cool energy demand of the unit. Negative values indicate that heating energy is required (or in use) by the boiler controller.
12 "Supply temperature output" (nvoSupplyTemp)	Optional	SNVT_temp_p	output	Used to monitor the supply water temperature that the boiler controller is using for control
13 "Local supply temperature output" (nvoLocalSupTemp)	Optional	SNVT_temp_p	output	Present value of a locally wired supply water temperature sensor

NV Member	Requirement	CNP Type	Direction	Description
14 "Return temperature output" (nvoReturnTemp)	Optional	SNVT_temp_p	output	Used to monitor the return water temperature that the boiler controller is using for control
15 "Local return temperature output" (nvoLocalRetTemp)	Optional	SNVT_temp_p	output	Present value of a locally wired return water temperature sensor
16 "Pump speed output" (nvoPumpSpeed)	Optional	SNVT_switch	output	Actual pump speed of a local multi-speed pump, as well as the requested speed of a remote pump
17 "Bypass valve output" (nvoBypassValve)	Optional	SNVT_lev_percent	output	Present value of the bypass control valve position (if hardwired) or can be used to control a remote bypass valve
18 "Outdoor air temperature output" (nvoOutdoorTemp)	Optional	SNVT_temp_p	output	Used to monitor the outdoor air temperature that the boiler controller is using for control
19 "Local outdoor air temperature output" (nvoLocalOATemp)	Optional	SNVT_temp_p	output	Present value of a locally wired outdoor air temperature sensor

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to SFPTboilerController	Mandatory	none	not permitted	Maximum period of time that expires before the specified network variable outputs will automatically be updated
SCPTpwrUpState applying to SFPTboilerController	Optional	none	not permitted	Default power-up and restart modes of the boiler controller
SCPTmaxRcvTime applying to SFPTboilerController	Optional	none	not permitted	Maximum time that elapses after the last update to a specified network variable input before the boiler controller starts to use its default values

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTheatSetpt applying to SFPTboilerController	Mandatory	none	not permitted	This applies to the setpoint NV input if one exists; otherwise, it applies to the hardwired input
SCPTlocation applying to SFPTboilerController	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTminSendTime applying to SFPTboilerController	Optional	none	not permitted	Minimum period of time between automatic network variable output transmissions
SCPTpumpDownDelay applying to SFPTboilerController	Optional	none	not permitted	Used to control the time delay that the pump remains "on" after a new command to turn the pump "off" is issued

7.66 SFPTspaceComfortController (8500)

NV Member	Requirement	CNP Type	Direction	Description
1 "Space temperature input" (nviSpaceTemp)	Mandatory	SNVT_temp_p	input	Connects an external space temperature sensor to the node
2 "Temperature setpoint input (absolute)" (nviSetpoint)	Optional	SNVT_temp_p	input	Allows the temperature setpoints for the occupied and standby modes to be changed via the network
3 "Setpoint offset input" (nviSetptOffset)	Optional	SNVT_temp_p	input	Shifts the effective occupied and standby temperature setpoints by adding this NV's value to the present setpoints
4 "Setpoint shift input" (nviSetptShift)	Optional	SNVT_temp_setpt	input	Shifts the effective heat/cool setpoints by adding the corresponding value in this NV to the present setpoints

NV Member	Requirement	CNP Type	Direction	Description
5 "Occupancy scheduler input" (nviOccSchedule)	Optional	SNVT_tod_event	input	Commands the SCC into different occupancy modes
6 "Occupancy override input" (nviOccManCmd)	Optional	SNVT_occupancy	input	Commands the SCC into different occupancy modes
7 "Occupancy sensor input" (nviOccSensor)	Optional	SNVT_occupancy	input	Indicates the presence of occupants in the controlled space
8 "Application mode input" (nviApplicMode)	Optional	SNVT_hvac_mode	input	Used to coordinate the SCC with any supervisory controller
9 "Heat/cool mode input" (nviHeatCool)	Optional	SNVT_hvac_mode	input	Used to coordinate the SCC with any node that may need to control the heat/cool changeover of the unit
10 "Fan speed command input" (nviFanSpeedCmd)	Optional	SNVT_switch	input	Enables connection of an external fan speed switch, or allows a supervisory device to override the fan speed
11 "Compressor enable input" (nviComprEnable)	Optional	SNVT_switch	input	This input is used to disable compressor operation
12 "Auxiliary heat enable input" (nviAuxHeatEnable)	Optional	SNVT_switch	input	This input is used to disable auxiliary heat operation
13 "Economizer enable input" (nviEconEnable)	Optional	SNVT_switch	input	This input is used to enable and disable economizer operation
14 "Energy hold-off input" (nviEnergyHoldOff)	Optional	SNVT_switch	input	This input is used to stop heating and cooling while allowing the unit to protect the space from temperature extremes

NV Member	Requirement	CNP Type	Direction	Description
15 "Water valve override input" (nviValveOverride)	Optional	SNVT_hvac_overid	input	Commands the controller into a manual mode for overriding water valves
16 "Airflow override input" (nviFlowOverride)	Optional	SNVT_hvac_overid	input	Commands the controller into a manual mode for overriding airflow control
17 "Emergency override input" (nviEmergOverride)	Optional	SNVT_hvac_emerg	input	Commands the device into different emergency modes
18 "Source temperature input" (nviSourceTemp)	Optional	SNVT_temp_p	input	Indicates the temperature of the air or water being supplied to the unit for heating and/or cooling capacity
19 "Outdoor air temperature input" (nviOutdoorTemp)	Optional	SNVT_temp_p	input	Represents information from an outdoor air temperature sensor
20 "Space humidity input" (nviSpaceRH)	Optional	SNVT_lev_percent	input	Measured space humidity in percent
21 "Outdoor air humidity input" (nviOutdoorRH)	Optional	SNVT_lev_percent	input	Measured outdoor humidity in percent
22 "Space indoor air quality input" (nviSpaceIAQ)	Optional	SNVT_ppm	input	Measured space CO2 or VOC levels in PPM
23 "Space dewpoint temperature input" (nviSpaceDewPt)	Optional	SNVT_temp_p	input	Measured space dewpoint temperature
24 "Outdoor air dewpoint temperature input" (nviOutdoorDewPt)	Optional	SNVT_temp_p	input	Measured outdoor dewpoint temperature

NV Member	Requirement	CNP Type	Direction	Description
25 "Airflow input" (nviAirflow)	Optional	SNVT_flow	input	The measured supply airflow value is typically provided by a flow sensor on the network
26 "Effective space temperature output" (nvoSpaceTemp)	Mandatory	SNVT_temp_p	output	Used to monitor the effective space temperature that the SCC is using for control
27 "Unit status output" (nvoUnitStatus)	Mandatory	SNVT_hvac_status	output	Reports the SCC status
28 "Effective setpoint output" (nvoEffectSetpt)	Optional	SNVT_temp_p	output	Monitors the effective temperature setpoint
29 "Effective occupancy output" (nvoEffectOccup)	Optional	SNVT_occupancy	output	Actual occupancy mode of the unit
30 "Effective heat/cool output" (nvoHeatCool)	Optional	SNVT_hvac_mode	output	Actual heat/cool mode of the unit
31 "Local setpoint output" (nvoSetpoint)	Optional	SNVT_temp_p	output	Space temperature setpoint value if a setpoint device is hardwired
32 "Local setpoint shift output" (nvoSetptShift)	Optional	SNVT_temp_setpt	output	Locally determined shift of the effective heat/cool setpoints
33 "Fan speed output" (nvoFanSpeed)	Optional	SNVT_switch	output	Actual fan speed of a local multi-speed fan as well as the requested speed of a remote fan
34 "Discharge air temperature output" (nvoDischAirTemp)	Optional	SNVT_temp_p	output	Monitors the temperature of the air that leaves the SCC
35 "Absolute power consumption output" (nvoLoadAbs)	Optional	SNVT_power	output	Present power consumption of the unit

NV Member	Requirement	CNP Type	Direction	Description
36 "Absolute power consumption (kW) output" (nvoLoadAbsK)	Optional	SNVT_power_kilo	output	Present power consumption of the unit
37 "Terminal load output" (nvoTerminalLoad)	Optional	SNVT_lev_percent	output	Present heat/cool energy demand of the unit
38 "Primary heat output" (nvoHeatPrimary)	Optional	SNVT_lev_percent	output	Present level of the primary heat output
39 "Secondary heat output" (nvoHeatSecondary)	Optional	SNVT_lev_percent	output	Present level of the secondary heat output
40 "Primary cool output" (nvoCoolPrimary)	Optional	SNVT_lev_percent	output	Present level of the primary mechanical cooling output
41 "Secondary cool output" (nvoCoolSecondary)	Optional	SNVT_lev_percent	output	Present level of the secondary mechanical cooling output
42 "Outdoor air damper output" (nvoOADamper)	Optional	SNVT_lev_percent	output	Present position of the outdoor air damper (if hardwired) or as a request to a remote outdoor air damper
43 "Space humidity output" (nvoSpaceRH)	Optional	SNVT_lev_percent	output	Space humidity in percent, if the SCC Device has a hardwired humidity sensor
44 "Outdoor air humidity output" (nvoOutdoorRH)	Optional	SNVT_lev_percent	output	Outdoor air humidity in percent, if the SCC Device has a hardwired humidity sensor
45 "Outdoor air temperature output" (nvoOutdoorTemp)	Optional	SNVT_temp_p	output	Monitors the outdoor air temperature if the unit controller provides a hardwired temperature sensor

NV Member	Requirement	CNP Type	Direction	Description
46 "Space CO2 sensor output" (nvoSpaceCO2)	Optional	SNVT_ppm	output	Space CO2 concentration in ppm, if the SCC Device has a hardwired CO2 sensor
47 "Space dewpoint temperature output" (nvoSpaceDewPt)	Optional	SNVT_temp_p	output	Space dewpoint temperature
48 "Humidifier output" (nvoHumidifier)	Optional	SNVT_lev_percent	output	Present value of the humidifier (if hardwired) or can be used to control a remote humidifier or control valve
49 "Energy hold-off output" (nvoEnergyHoldOff)	Optional	SNVT_switch	output	Present state of an energy hold-off device that is hardwired to the controller
50 "Effective airflow setpoint output" (nvoEffectFlowSP)	Optional	SNVT_flow	output	Active flow setpoint used by the flow control loop
51 "Flow control damper setpoint output" (nvoFlowSetpoint)	Optional	SNVT_lev_percent	output	Active flow setpoint used by the flow control loop
52 "Airflow output" (nvoAirflow)	Optional	SNVT_flow	output	Measured airflow in the unit
53 "Heat source temperature input" (nviHeatSrcTemp)	Optional	SNVT_temp_p	input	Temperature of the air or water being supplied to the unit for heating capacity
54 "Cool source temperature input" (nviCoolSrcTemp)	Optional	SNVT_temp_p	input	Temperature of the air or water being supplied to the unit for cooling capacity
55 "Primary heat input for slave operation" (nviHeatPriSlave)	Optional	SNVT_lev_percent	input	This input NV is intended for slave operation

NV Member	Requirement	CNP Type	Direction	Description
56 "Secondary heat input for slave operation" (nviHeatSecSlave)	Optional	SNVT_lev_percent	input	This input NV is intended for slave operation
57 "Primary cool input for slave operation" (nviCoolPriSlave)	Optional	SNVT_lev_percent	input	This input NV is intended for slave operation
58 "Secondary cool input for slave operation" (nviCoolSecSlave)	Optional	SNVT_lev_percent	input	This input NV is intended for slave operation
59 "Minimum position OA damper input" (nviOAMinPos)	Optional	SNVT_lev_percent	input	Dynamic minimum position setpoint for an outdoor air damper. When valid it will supersede nciOAMinPos.
60 "Minimum air flow setpoint input" (nviMinAirFlow)	Optional	SNVT_lev_percent	input	Dynamic minimum cooling air flow setpoint for single or dual duct VAV terminal units. When valid it will supersede nciMinFlow.
61 "Minimum heat air flow setpoint input" (nviMinAirFlowHt)	Optional	SNVT_lev_percent	input	Dynamic minimum heating air flow setpoint for single or dual duct VAV terminal units. When valid it will supersede nciMinFlowHeat.
62 "Air flow setpoint input" (nviAirFlowSetpt)	Optional	SNVT_flow	input	The controller will add this input with nciFlowOffset to derive the active flow setpoint
63 "Terminal load input" (nviTerminalLoad)	Optional	SNVT_lev_percent	input	When used with nvoTerminalLoad from another controller can be used to coordinate master/salve operation
64 "Mixed air temperature output" (nvoMixedAirTemp)	Optional	SNVT_temp_p	output	The temperature of the combined return and fresh airstreams in an AHU before they reach the water coils
65 "Local space temperature output" (nvoLocalSpaceTmp)	Optional	SNVT_temp_p	output	Local hardwired space temperature input

NV Member	Requirement	CNP Type	Direction	Description
66 "Effective air flow heat setpoint output" (nvoEffFlowSPHeat)	Optional	SNVT_flow	output	The hot or ventilation duct flow setpoint of a dual duct unit.
67 "Flow control damper heat setpoint output" (nvoFlowSPHeat)	Optional	SNVT_lev_percent	output	Active flow setpoint used by the flow control loop for a hot or ventilation deck in a dual duct unit
68 "Air flow heat output" (nvoAirFlowHeat)	Optional	SNVT_flow	output	Air flow of a hot or ventilation deck of a dual duct VAV terminal
69 "HVAC saturation status" (nvoSatStatus)	Optional	SNVT_hvac_satsts	output	Indicates whether the control algorithm capacity limits, or end device physical limits, have been reached

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to SFPTspaceComfortController	Mandatory	none	not permitted	Maximum period of time that expires before the specified NV outputs will automatically be updated
SCPTsetPnts applying to SFPTspaceComfortController	Mandatory	none	not permitted	Space temperature setpoints for the various heat, cool, and occupancy modes
SCPTminSendTime applying to SFPTspaceComfortController	Optional	none	not permitted	Minimum period of time between automatic NV output transmissions
SCPTmaxRcvTime applying to SFPTspaceComfortController	Optional	none	not permitted	Maximum time that elapses after the last update to a specified NV input before the SCC starts to use its default values
SCPTlocation applying to SFPTspaceComfortController	Optional	none	not permitted	Provides descriptive physical location information related to the object.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTbypassTime applying to nviOccManCmd	Optional	none	not permitted	Maximum amount of time that the SCC can be in the bypass (occupancy) mode following a single bypass request
SCPTmanOvrTime applying to nviValveOverride	Optional	none	not permitted	Maximum time that the SCC will stay in a manual mode that was requested by a NV input, without receiving an update on that NV
SCPTminRnge applying to nvoOADamper	Optional	none	not permitted	Minimum position for the outdoor air damper
SCPTlimitCO2 applying to nviSpaceIAQ	Optional	none	not permitted	Maximum limit to allowable carbon dioxide within a defined area
SCPThumSetpt applying to nviSpaceRH	Optional	none	not permitted	High-limit humidity setpoint for the controlled space
SCPTnumValves applying to SFPTspaceComfortController	Optional	none	not permitted	Selects whether the SCC is used in a two pipe (one valve) or four pipe (two valve) system
SCPTductArea applying to SFPTspaceComfortController	Optional	none	not permitted	Nominal cross-sectional airflow area of a VAV terminal
SCPTnomAirFlow applying to nvoFlowSetpoint	Optional	none	not permitted	Nominal airflow volume of a VAV terminal
SCPTsensConstVAV applying to SFPTspaceComfortController	Optional	none	not permitted	Calibrates the airflow reading of a VAV terminal
SCPTminFlow applying to nviAirflow	Optional	none	not permitted	Minimum airflow setpoint of a VAV terminal
SCPTmaxFlow applying to nviAirflow	Optional	none	not permitted	Maximum airflow setpoint of a VAV terminal

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTminFlowHeat applying to SFPTspaceComfortController	Optional	none	not permitted	Minimum airflow setpoint of a VAV terminal while heating
SCPTmaxFlowHeat applying to SFPTspaceComfortController	Optional	none	not permitted	Maximum airflow setpoint of a VAV terminal while heating
SCPTminFlowStby applying to SFPTspaceComfortController	Optional	none	not permitted	Minimum airflow setpoint of a VAV terminal in the Standby (occupancy) mode
SCPThvacType applying to SFPTspaceComfortController	Mandatory	device specific, constant	not permitted	This value is set by the manufacturer to allow an integrator to know the function of this SCC device
SCPTfanOperation applying to SFPTspaceComfortController	Optional	none	not permitted	Specifies fan operation during occupied and occupied standby. Fan operation during unoccupied is manufacturer defined
SCPTminFlowUnit applying to SFPTspaceComfortController	Optional	none	not permitted	Unit minimum air flow for dual duct VAV Terminal units
SCPTmaxFlowUnit applying to SFPTspaceComfortController	Optional	none	not permitted	Unit maximum air flow for dual duct VAV Terminal units
SCPTminFlowHeatStby applying to SFPTspaceComfortController	Optional	none	not permitted	The heating or ventilated deck minimum flow of a dual duct VAV Terminal unit during occupied standby mode
SCPTminFlowUnitStby applying to SFPTspaceComfortController	Optional	none	not permitted	Total unit minimum airflow for dual duct units during occupied standby mode
SCPToffsetFlow applying to SFPTspaceComfortController	Optional	none	not permitted	

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTareaDuctHeat applying to SFPTspaceComfortController	Optional	none	not permitted	Nominal cross-sectional airflow area of the hot or ventilation deck of a dual duct VAV terminal unit
SCPTnomAirFlowHeat applying to SFPTspaceComfortController	Optional	none	not permitted	Value used to provide the nominal airflow volume of a hot or ventilation deck of a dual duct VAV terminal
SCPTgainVAVHeat applying to SFPTspaceComfortController	Optional	none	not permitted	Calibration constant used to calculate airflow
SCPTnumDampers applying to SFPTspaceComfortController	Optional	none	not permitted	Indicates to the controller if it is in a single or dual duct system
SCPTminFlowUnitHeat applying to SFPTspaceComfortController	Optional	none	not permitted	Minimum airflow setpoint of a single duct, or the unit minimum airflow setpoint of a dual duct VAV terminal when using a unit (local) heating source
SCPTsaturationDelay applying to SFPTspaceComfortController	Optional	none	not permitted	

7.67 SFPTsccFanCoil (8501)

NV Member	Requirement	CNP Type	Direction	Description
1 "Space temperature input" (nviSpaceTemp)	Mandatory	SNVT_temp_p	input	Connects an external space temperature sensor to the node
2 "Temperature setpoint input (absolute)" (nviSetpoint)	Optional	SNVT_temp_p	input	Allows the temperature setpoints for the occupied and standby modes to be changed via the network

NV Member	Requirement	CNP Type	Direction	Description
3 "Setpoint offset input" (nviSetptOffset)	Optional	SNVT_temp_p	input	Shifts the effective occupied and standby temperature setpoints by adding this NV's value to the present setpoints
4 "Setpoint shift input" (nviSetptShift)	Optional	SNVT_temp_setpt	input	Shifts the effective heat/cool setpoints by adding the corresponding value in this NV to the present setpoints
5 "Occupancy scheduler input" (nviOccSchedule)	Optional	SNVT_tod_event	input	Commands the SCC into different occupancy modes
6 "Occupancy override input" (nviOccManCmd)	Optional	SNVT_occupancy	input	Commands the SCC into different occupancy modes
7 "Occupancy sensor input" (nviOccSensor)	Optional	SNVT_occupancy	input	Indicates the presence of occupants in the controlled space
8 "Application mode input" (nviApplicMode)	Optional	SNVT_hvac_mode	input	Used to coordinate the SCC with any supervisory controller
9 "Heat/cool mode input" (nviHeatCool)	Optional	SNVT_hvac_mode	input	Used to coordinate the SCC with any node that may need to control the heat/cool changeover of the unit
10 "Fan speed command input" (nviFanSpeedCmd)	Optional	SNVT_switch	input	Enables connection of an external fan speed switch, or allows a supervisory device to override the fan speed
11 "Compressor enable input" (nviComprEnable)	Optional	SNVT_switch	input	This input is used to disable compressor operation
12 "Auxiliary heat enable input" (nviAuxHeatEnable)	Optional	SNVT_switch	input	This input is used to disable auxiliary heat operation

NV Member	Requirement	CNP Type	Direction	Description
13 "Economizer enable input" (nviEconEnable)	Optional	SNVT_switch	input	This input is used to enable and disable economizer operation
14 "Energy hold-off input" (nviEnergyHoldOff)	Optional	SNVT_switch	input	This input is used to stop heating and cooling while allowing the unit to protect the space from temperature extremes
15 "Water valve override input" (nviValveOverride)	Optional	SNVT_hvac_overid	input	Commands the controller into a manual mode for overriding water valves
16 "Airflow override input" (nviFlowOverride)	Optional	SNVT_hvac_overid	input	Commands the controller into a manual mode for overriding airflow control
17 "Emergency override input" (nviEmergOverride)	Optional	SNVT_hvac_emerg	input	Commands the device into different emergency modes
18 "Source temperature input" (nviSourceTemp)	Optional	SNVT_temp_p	input	Indicates the temperature of the air or water being supplied to the unit for heating and/or cooling capacity
19 "Outdoor air temperature input" (nviOutdoorTemp)	Optional	SNVT_temp_p	input	Represents information from an outdoor air temperature sensor
20 "Space humidity input" (nviSpaceRH)	Optional	SNVT_lev_percent	input	Measured space humidity in percent
21 "Outdoor air humidity input" (nviOutdoorRH)	Optional	SNVT_lev_percent	input	Measured outdoor humidity in percent
22 "Space indoor air quality input" (nviSpaceIAQ)	Optional	SNVT_ppm	input	Measured space CO2 or VOC levels in PPM

NV Member	Requirement	CNP Type	Direction	Description
23 "Space dewpoint temperature input" (nviSpaceDewPt)	Optional	SNVT_temp_p	input	Measured space dewpoint temperature
24 "Outdoor air dewpoint temperature input" (nviOutdoorDewPt)	Optional	SNVT_temp_p	input	Measured outdoor dewpoint temperature
25 "Airflow input" (nviAirflow)	Optional	SNVT_flow	input	The measured supply airflow value is typically provided by a flow sensor on the network
26 "Effective space temperature output" (nvoSpaceTemp)	Mandatory	SNVT_temp_p	output	Used to monitor the effective space temperature that the SCC is using for control
27 "Unit status output" (nvoUnitStatus)	Mandatory	SNVT_hvac_status	output	Reports the SCC status
28 "Effective setpoint output" (nvoEffectSetpt)	Optional	SNVT_temp_p	output	Monitors the effective temperature setpoint
29 "Effective occupancy output" (nvoEffectOccup)	Optional	SNVT_occupancy	output	Actual occupancy mode of the unit
30 "Effective heat/cool output" (nvoHeatCool)	Optional	SNVT_hvac_mode	output	Actual heat/cool mode of the unit
31 "Local setpoint output" (nvoSetpoint)	Optional	SNVT_temp_p	output	Space temperature setpoint value if a setpoint device is hardwired
32 "Local setpoint shift output" (nvoSetptShift)	Optional	SNVT_temp_setpt	output	Locally determined shift of the effective heat/cool setpoints

NV Member	Requirement	CNP Type	Direction	Description
33 "Fan speed output" (nvoFanSpeed)	Optional	SNVT_switch	output	Actual fan speed of a local multi-speed fan as well as the requested speed of a remote fan
34 "Discharge air temperature output" (nvoDischAirTemp)	Optional	SNVT_temp_p	output	Monitors the temperature of the air that leaves the SCC
35 "Absolute power consumption output" (nvoLoadAbs)	Optional	SNVT_power	output	Present power consumption of the unit
36 "Absolute power consumption (kW) output" (nvoLoadAbsK)	Optional	SNVT_power_kilo	output	Present power consumption of the unit
37 "Terminal load output" (nvoTerminalLoad)	Optional	SNVT_lev_percent	output	Present heat/cool energy demand of the unit
38 "Primary heat output" (nvoHeatPrimary)	Optional	SNVT_lev_percent	output	Present level of the primary heat output
39 "Secondary heat output" (nvoHeatSecondary)	Optional	SNVT_lev_percent	output	Present level of the secondary heat output
40 "Primary cool output" (nvoCoolPrimary)	Optional	SNVT_lev_percent	output	Present level of the primary mechanical cooling output
41 "Secondary cool output" (nvoCoolSecondary)	Optional	SNVT_lev_percent	output	Present level of the secondary mechanical cooling output
42 "Outdoor air damper output" (nvoOADamper)	Optional	SNVT_lev_percent	output	Present position of the outdoor air damper (if hardwired) or as a request to a remote outdoor air damper

NV Member	Requirement	CNP Type	Direction	Description
43 "Space humidity output" (nvoSpaceRH)	Optional	SNVT_lev_percent	output	Space humidity in percent, if the SCC Device has a hardwired humidity sensor
44 "Outdoor air humidity output" (nvoOutdoorRH)	Optional	SNVT_lev_percent	output	Outdoor air humidity in percent, if the SCC Device has a hardwired humidity sensor
45 "Outdoor air temperature output" (nvoOutdoorTemp)	Optional	SNVT_temp_p	output	Monitors the outdoor air temperature if the unit controller provides a hardwired temperature sensor
46 "Space CO2 sensor output" (nvoSpaceCO2)	Optional	SNVT_ppm	output	Space CO2 concentration in ppm, if the SCC Device has a hardwired CO2 sensor
47 "Space dewpoint temperature output" (nvoSpaceDewPt)	Optional	SNVT_temp_p	output	Space dewpoint temperature
48 "Humidifier output" (nvoHumidifier)	Optional	SNVT_lev_percent	output	Present value of the humidifier (if hardwired) or can be used to control a remote humidifier or control valve
49 "Energy hold-off output" (nvoEnergyHoldOff)	Optional	SNVT_switch	output	Present state of an energy hold-off device that is hardwired to the controller
50 "Effective airflow setpoint output" (nvoEffectFlowSP)	Optional	SNVT_flow	output	Active flow setpoint used by the flow control loop
51 "Flow control damper setpoint output" (nvoFlowSetpoint)	Optional	SNVT_lev_percent	output	Active flow setpoint used by the flow control loop
52 "Airflow output" (nvoAirflow)	Optional	SNVT_flow	output	Measured airflow in the unit

NV Member	Requirement	CNP Type	Direction	Description
53 "Heat source temperature input" (nviHeatSrcTemp)	Optional	SNVT_temp_p	input	Temperature of the air or water being supplied to the unit for heating capacity
54 "Cool source temperature input" (nviCoolSrcTemp)	Optional	SNVT_temp_p	input	Temperature of the air or water being supplied to the unit for cooling capacity
55 "Primary heat input for slave operation" (nviHeatPriSlave)	Optional	SNVT_lev_percent	input	This input NV is intended for slave operation
56 "Secondary heat input for slave operation" (nviHeatSecSlave)	Optional	SNVT_lev_percent	input	This input NV is intended for slave operation
57 "Primary cool input for slave operation" (nviCoolPriSlave)	Optional	SNVT_lev_percent	input	This input NV is intended for slave operation
58 "Secondary cool input for slave operation" (nviCoolSecSlave)	Optional	SNVT_lev_percent	input	This input NV is intended for slave operation
59 "Minimum position OA damper input" (nviOAMinPos)	Optional	SNVT_lev_percent	input	Dynamic minimum position setpoint for an outdoor air damper. When valid it will supersede nciOAMinPos.
60 "Minimum air flow setpoint input" (nviMinAirFlow)	Optional	SNVT_lev_percent	input	Dynamic minimum cooling airflow setpoint for single or dual duct VAV terminal units. When valid it will supersede nciMinFlow.
61 "Minimum heat air flow setpoint input" (nviMinAirFlowHt)	Optional	SNVT_lev_percent	input	Dynamic minimum heating airflow setpoint for single or dual duct VAV terminal units. When valid it will supersede nciMinFlowHeat.
62 "Air flow setpoint input" (nviAirFlowSetpt)	Optional	SNVT_flow	input	The controller will add this input with nciFlowOffset to derive the active flow setpoint

NV Member	Requirement	CNP Type	Direction	Description
63 "Terminal load input" (nviTerminalLoad)	Optional	SNVT_lev_percent	input	When used with nvoTerminalLoad from another controller can be used to coordinate master/salve operation
64 "Mixed air temperature output" (nvoMixedAirTemp)	Optional	SNVT_temp_p	output	The temperature of the combined return and fresh airstreams in an AHU before they reach the water coils
65 "Local space temperature output" (nvoLocalSpaceTmp)	Optional	SNVT_temp_p	output	Local hardwired space temperature input
66 "Effective air flow heat setpoint output" (nvoEffFlowSPHeat)	Optional	SNVT_flow	output	The hot or ventilation duct flow setpoint of a dual duct unit.
67 "Flow control damper heat setpoint output" (nvoFlowSPHeat)	Optional	SNVT_lev_percent	output	Active flow setpoint used by the flow control loop for a hot or ventilation deck in a dual duct unit
68 "Air flow heat output" (nvoAirFlowHeat)	Optional	SNVT_flow	output	Air flow of a hot or ventilation deck of a dual duct VAV terminal
69 "HVAC saturation status" (nvoSatStatus)	Optional	SNVT_hvac_satsts	output	Indicates whether the control algorithm capacity limits, or end device physical limits, have been reached

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to SFPTsccFanCoil	Mandatory	none	not permitted	Maximum period of time that expires before the specified NV outputs will automatically be updated
SCPTsetPnts applying to SFPTsccFanCoil	Mandatory	none	not permitted	Space temperature setpoints for the various heat, cool, and occupancy modes
SCPTminSendTime applying to SFPTsccFanCoil	Optional	none	not permitted	Minimum period of time between automatic NV output transmissions

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxRcvTime applying to SFPTsccFanCoil	Optional	none	not permitted	Maximum time that elapses after the last update to a specified NV input before the SCC starts to use its default values
SCPTlocation applying to SFPTsccFanCoil	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTbypassTime applying to nviOccManCmd	Optional	none	not permitted	Maximum amount of time that the SCC can be in the bypass (occupancy) mode following a single bypass request
SCPTmanOvrTime applying to nviValveOverride	Optional	none	not permitted	Maximum time that the SCC will stay in a manual mode that was requested by a NV input, without receiving an update on that NV
SCPTminRnge applying to nvoOADamper	Optional	none	not permitted	Minimum position for the outdoor air damper
SCPTlimitCO2 applying to nviSpaceIAQ	Optional	none	not permitted	Maximum limit to allowable carbon dioxide within a defined area
SCPThumSetpt applying to nviSpaceRH	Optional	none	not permitted	High-limit humidity setpoint for the controlled space
SCPTnumValves applying to SFPTsccFanCoil	Optional	none	not permitted	Selects whether the SCC is used in a two pipe (one valve) or four pipe (two valve) system
SCPTductArea applying to SFPTsccFanCoil	Optional	none	not permitted	Nominal cross-sectional airflow area of a VAV terminal
SCPTnomAirFlow applying to nvoFlowSetpoint	Optional	none	not permitted	Nominal airflow volume of a VAV terminal
SCPTsensConstVAV applying to SFPTsccFanCoil	Optional	none	not permitted	Calibrates the airflow reading of a VAV terminal
SCPTminFlow applying to nviAirflow	Optional	none	not permitted	Minimum airflow setpoint of a VAV terminal

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxFlow applying to nviAirflow	Optional	none	not permitted	Maximum airflow setpoint of a VAV terminal
SCPTminFlowHeat applying to SFPTsccFanCoil	Optional	none	not permitted	Minimum airflow setpoint of a VAV terminal while heating
SCPTmaxFlowHeat applying to SFPTsccFanCoil	Optional	none	not permitted	Maximum airflow setpoint of a VAV terminal while heating
SCPTminFlowStby applying to SFPTsccFanCoil	Optional	none	not permitted	Minimum airflow setpoint of a VAV terminal in the Standby (occupancy) mode
SCPThvacType applying to SFPTsccFanCoil	Optional	device specific, constant	not permitted	This value is set by the manufacturer to allow an integrator to know the function of this SCC device
SCPTfanOperation applying to SFPTsccFanCoil	Optional	none	not permitted	Specifies fan operation during occupied and occupied standby. Fan operation during unoccupied is manufacturer defined
SCPTminFlowUnit applying to SFPTsccFanCoil	Optional	none	not permitted	Unit minimum air flow for dual duct VAV Terminal units
SCPTmaxFlowUnit applying to SFPTsccFanCoil	Optional	none	not permitted	Unit maximum air flow for dual duct VAV Terminal units
SCPTminFlowHeatStby applying to SFPTsccFanCoil	Optional	none	not permitted	The heating or ventilated deck minimum flow of a dual duct VAV Terminal unit during occupied standby mode
SCPTminFlowUnitStby applying to SFPTsccFanCoil	Optional	none	not permitted	Total unit minimum airflow for dual duct units during occupied standby mode
SCPToffsetFlow applying to SFPTsccFanCoil	Optional	none	not permitted	
SCPTareaDuctHeat applying to SFPTsccFanCoil	Optional	none	not permitted	Nominal cross-sectional airflow area of the hot or ventilation deck of a dual duct VAV terminal unit

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTnomAirFlowHeat applying to SFPTsccFanCoil	Optional	none	not permitted	Value used to provide the nominal airflow volume of a hot or ventilation deck of a dual duct VAV terminal
SCPTgainVAVHeat applying to SFPTsccFanCoil	Optional	none	not permitted	Calibration constant used to calculate airflow
SCPTnumDampers applying to SFPTsccFanCoil	Optional	none	not permitted	Indicates to the controller if it is in a single or dual duct system
SCPTminFlowUnitHeat applying to SFPTsccFanCoil	Optional	none	not permitted	Minimum airflow setpoint of a single duct, or the unit minimum airflow setpoint of a dual duct VAV terminal when using a unit (local) heating source
SCPTsaturationDelay applying to SFPTsccFanCoil	Optional	none	not permitted	

7.68 SFPTsccVAV (8502)

NV Member	Requirement	CNP Type	Direction	Description
1 "Space temperature input" (nviSpaceTemp)	Mandatory	SNVT_temp_p	input	Connects an external space temperature sensor to the node
2 "Temperature setpoint input (absolute)" (nviSetpoint)	Optional	SNVT_temp_p	input	Allows the temperature setpoints for the occupied and standby modes to be changed via the network
3 "Setpoint offset input" (nviSetptOffset)	Optional	SNVT_temp_p	input	Shifts the effective occupied and standby temperature setpoints by adding this NV's value to the present setpoints
4 "Setpoint shift input" (nviSetptShift)	Optional	SNVT_temp_setpt	input	Shifts the effective heat/cool setpoints by adding the corresponding value in this NV to the present setpoints

NV Member	Requirement	CNP Type	Direction	Description
5 "Occupancy scheduler input" (nviOccSchedule)	Optional	SNVT_tod_event	input	Commands the SCC into different occupancy modes
6 "Occupancy override input" (nviOccManCmd)	Optional	SNVT_occupancy	input	Commands the SCC into different occupancy modes
7 "Occupancy sensor input" (nviOccSensor)	Optional	SNVT_occupancy	input	Indicates the presence of occupants in the controlled space
8 "Application mode input" (nviApplicMode)	Optional	SNVT_hvac_mode	input	Used to coordinate the SCC with any supervisory controller
9 "Heat/cool mode input" (nviHeatCool)	Optional	SNVT_hvac_mode	input	Used to coordinate the SCC with any node that may need to control the heat/cool changeover of the unit
10 "Fan speed command input" (nviFanSpeedCmd)	Optional	SNVT_switch	input	Enables connection of an external fan speed switch, or allows a supervisory device to override the fan speed
11 "Compressor enable input" (nviComprEnable)	Optional	SNVT_switch	input	This input is used to disable compressor operation
12 "Auxiliary heat enable input" (nviAuxHeatEnable)	Optional	SNVT_switch	input	This input is used to disable auxiliary heat operation
13 "Economizer enable input" (nviEconEnable)	Optional	SNVT_switch	input	This input is used to enable and disable economizer operation
14 "Energy hold-off input" (nviEnergyHoldOff)	Optional	SNVT_switch	input	This input is used to stop heating and cooling while allowing the unit to protect the space from temperature extremes

NV Member	Requirement	CNP Type	Direction	Description
15 "Water valve override input" (nviValveOverride)	Optional	SNVT_hvac_overid	input	Commands the controller into a manual mode for overriding water valves
16 "Airflow override input" (nviFlowOverride)	Optional	SNVT_hvac_overid	input	Commands the controller into a manual mode for overriding airflow control
17 "Emergency override input" (nviEmergOverride)	Optional	SNVT_hvac_emerg	input	Commands the device into different emergency modes
18 "Source temperature input" (nviSourceTemp)	Optional	SNVT_temp_p	input	Indicates the temperature of the air or water being supplied to the unit for heating and/or cooling capacity
19 "Outdoor air temperature input" (nviOutdoorTemp)	Optional	SNVT_temp_p	input	Represents information from an outdoor air temperature sensor
20 "Space humidity input" (nviSpaceRH)	Optional	SNVT_lev_percent	input	Measured space humidity in percent
21 "Outdoor air humidity input" (nviOutdoorRH)	Optional	SNVT_lev_percent	input	Measured outdoor humidity in percent
22 "Space indoor air quality input" (nviSpaceIAQ)	Optional	SNVT_ppm	input	Measured space CO2 or VOC levels in PPM
23 "Space dewpoint temperature input" (nviSpaceDewPt)	Optional	SNVT_temp_p	input	Measured space dewpoint temperature
24 "Outdoor air dewpoint temperature input" (nviOutdoorDewPt)	Optional	SNVT_temp_p	input	Measured outdoor dewpoint temperature

NV Member	Requirement	CNP Type	Direction	Description
25 "Airflow input" (nviAirflow)	Optional	SNVT_flow	input	The measured supply airflow value is typically provided by a flow sensor on the network
26 "Effective space temperature output" (nvoSpaceTemp)	Mandatory	SNVT_temp_p	output	Used to monitor the effective space temperature that the SCC is using for control
27 "Unit status output" (nvoUnitStatus)	Mandatory	SNVT_hvac_status	output	Reports the SCC status
28 "Effective setpoint output" (nvoEffectSetpt)	Optional	SNVT_temp_p	output	Monitors the effective temperature setpoint
29 "Effective occupancy output" (nvoEffectOccup)	Optional	SNVT_occupancy	output	Actual occupancy mode of the unit
30 "Effective heat/cool output" (nvoHeatCool)	Optional	SNVT_hvac_mode	output	Actual heat/cool mode of the unit
31 "Local setpoint output" (nvoSetpoint)	Optional	SNVT_temp_p	output	Space temperature setpoint value if a setpoint device is hardwired
32 "Local setpoint shift output" (nvoSetptShift)	Optional	SNVT_temp_setpt	output	Locally determined shift of the effective heat/cool setpoints
33 "Fan speed output" (nvoFanSpeed)	Optional	SNVT_switch	output	Actual fan speed of a local multi-speed fan as well as the requested speed of a remote fan
34 "Discharge air temperature output" (nvoDischAirTemp)	Optional	SNVT_temp_p	output	Monitors the temperature of the air that leaves the SCC
35 "Absolute power consumption output" (nvoLoadAbs)	Optional	SNVT_power	output	Present power consumption of the unit

NV Member	Requirement	CNP Type	Direction	Description
36 "Absolute power consumption (kW) output" (nvoLoadAbsK)	Optional	SNVT_power_kilo	output	Present power consumption of the unit
37 "Terminal load output" (nvoTerminalLoad)	Optional	SNVT_lev_percent	output	Present heat/cool energy demand of the unit
38 "Primary heat output" (nvoHeatPrimary)	Optional	SNVT_lev_percent	output	Present level of the primary heat output
39 "Secondary heat output" (nvoHeatSecondary)	Optional	SNVT_lev_percent	output	Present level of the secondary heat output
40 "Primary cool output" (nvoCoolPrimary)	Optional	SNVT_lev_percent	output	Present level of the primary mechanical cooling output
41 "Secondary cool output" (nvoCoolSecondary)	Optional	SNVT_lev_percent	output	Present level of the secondary mechanical cooling output
42 "Outdoor air damper output" (nvoOADamper)	Optional	SNVT_lev_percent	output	Present position of the outdoor air damper (if hardwired) or as a request to a remote outdoor air damper
43 "Space humidity output" (nvoSpaceRH)	Optional	SNVT_lev_percent	output	Space humidity in percent, if the SCC Device has a hardwired humidity sensor
44 "Outdoor air humidity output" (nvoOutdoorRH)	Optional	SNVT_lev_percent	output	Outdoor air humidity in percent, if the SCC Device has a hardwired humidity sensor
45 "Outdoor air temperature output" (nvoOutdoorTemp)	Optional	SNVT_temp_p	output	Monitors the outdoor air temperature if the unit controller provides a hardwired temperature sensor

NV Member	Requirement	CNP Type	Direction	Description
46 "Space CO2 sensor output" (nvoSpaceCO2)	Optional	SNVT_ppm	output	Space CO2 concentration in ppm, if the SCC Device has a hardwired CO2 sensor
47 "Space dewpoint temperature output" (nvoSpaceDewPt)	Optional	SNVT_temp_p	output	Space dewpoint temperature
48 "Humidifier output" (nvoHumidifier)	Optional	SNVT_lev_percent	output	Present value of the humidifier (if hardwired) or can be used to control a remote humidifier or control valve
49 "Energy hold-off output" (nvoEnergyHoldOff)	Optional	SNVT_switch	output	Present state of an energy hold-off device that is hardwired to the controller
50 "Effective airflow setpoint output" (nvoEffectFlowSP)	Optional	SNVT_flow	output	Active flow setpoint used by the flow control loop
51 "Flow control damper setpoint output" (nvoFlowSetpoint)	Optional	SNVT_lev_percent	output	Active flow setpoint used by the flow control loop
52 "Airflow output" (nvoAirflow)	Optional	SNVT_flow	output	Measured airflow in the unit
53 "Heat source temperature input" (nviHeatSrcTemp)	Optional	SNVT_temp_p	input	Temperature of the air or water being supplied to the unit for heating capacity
54 "Cool source temperature input" (nviCoolSrcTemp)	Optional	SNVT_temp_p	input	Temperature of the air or water being supplied to the unit for cooling capacity
55 "Primary heat input for slave operation" (nviHeatPriSlave)	Optional	SNVT_lev_percent	input	This input NV is intended for slave operation

NV Member	Requirement	CNP Type	Direction	Description
56 "Secondary heat input for slave operation" (nviHeatSecSlave)	Optional	SNVT_lev_percent	input	This input NV is intended for slave operation
57 "Primary cool input for slave operation" (nviCoolPriSlave)	Optional	SNVT_lev_percent	input	This input NV is intended for slave operation
58 "Secondary cool input for slave operation" (nviCoolSecSlave)	Optional	SNVT_lev_percent	input	This input NV is intended for slave operation
59 "Minimum position OA damper input" (nviOAMinPos)	Optional	SNVT_lev_percent	input	Dynamic minimum position setpoint for an outdoor air damper. When valid it will supersede nciOAMinPos.
60 "Minimum air flow setpoint input" (nviMinAirFlow)	Optional	SNVT_lev_percent	input	Dynamic minimum cooling airflow setpoint for single or dual duct VAV terminal units. When valid it will supersede nciMinFlow.
61 "Minimum heat air flow setpoint input" (nviMinAirFlowHt)	Optional	SNVT_lev_percent	input	Dynamic minimum heating airflow setpoint for single or dual duct VAV terminal units. When valid it will supersede nciMinFlowHeat.
62 "Air flow setpoint input" (nviAirFlowSetpt)	Optional	SNVT_flow	input	The controller will add this input with nciFlowOffset to derive the active flow setpoint
63 "Terminal load input" (nviTerminalLoad)	Optional	SNVT_lev_percent	input	When used with nvoTerminalLoad from another controller can be used to coordinate master/salve operation
64 "Mixed air temperature output" (nvoMixedAirTemp)	Optional	SNVT_temp_p	output	The temperature of the combined return and fresh airstreams in an AHU before they reach the water coils
65 "Local space temperature output" (nvoLocalSpaceTmp)	Optional	SNVT_temp_p	output	Local hardwired space temperature input

NV Member	Requirement	CNP Type	Direction	Description
66 "Effective air flow heat setpoint output" (nvoEffFlowSPHeat)	Optional	SNVT_flow	output	The hot or ventilation duct flow setpoint of a dual duct unit.
67 "Flow control damper heat setpoint output" (nvoFlowSPHeat)	Optional	SNVT_lev_percent	output	Active flow setpoint used by the flow control loop for a hot or ventilation deck in a dual duct unit
68 "Air flow heat output" (nvoAirFlowHeat)	Optional	SNVT_flow	output	Air flow of a hot or ventilation deck of a dual duct VAV terminal
69 "HVAC saturation status" (nvoSatStatus)	Optional	SNVT_hvac_satsts	output	Indicates whether the control algorithm capacity limits, or end device physical limits, have been reached

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to SFPTsccVAV	Mandatory	none	not permitted	Maximum period of time that expires before the specified NV outputs will automatically be updated
SCPTsetPnts applying to SFPTsccVAV	Mandatory	none	not permitted	Space temperature setpoints for the various heat, cool, and occupancy modes
SCPTminSendTime applying to SFPTsccVAV	Optional	none	not permitted	Minimum period of time between automatic NV output transmissions
SCPTmaxRcvTime applying to SFPTsccVAV	Optional	none	not permitted	Maximum time that elapses after the last update to a specified NV input before the SCC starts to use its default values
SCPTlocation applying to SFPTsccVAV	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTbypassTime applying to nviOccManCmd	Optional	none	not permitted	Maximum amount of time that the SCC can be in the bypass (occupancy) mode following a single bypass request

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmanOvrTime applying to nviValveOverride	Optional	none	not permitted	Maximum time that the SCC will stay in a manual mode that was requested by a NV input, without receiving an update on that NV
SCPTminRnge applying to nvoOADamper	Optional	none	not permitted	Minimum position for the outdoor air damper
SCPTlimitCO2 applying to nviSpaceIAQ	Optional	none	not permitted	Maximum limit to allowable carbon dioxide within a defined area
SCPThumSetpt applying to nviSpaceRH	Optional	none	not permitted	High-limit humidity setpoint for the controlled space
SCPTnumValves applying to SFPTsccVAV	Optional	none	not permitted	Selects whether the SCC is used in a two pipe (one valve) or four pipe (two valve) system
SCPTductArea applying to SFPTsccVAV	Optional	none	not permitted	Nominal cross-sectional airflow area of a VAV terminal
SCPTnomAirFlow applying to nvoFlowSetpoint	Optional	none	not permitted	Nominal airflow volume of a VAV terminal
SCPTsensConstVAV applying to SFPTsccVAV	Optional	none	not permitted	Calibrates the airflow reading of a VAV terminal
SCPTminFlow applying to nviAirflow	Optional	none	not permitted	Minimum airflow setpoint of a VAV terminal
SCPTmaxFlow applying to nviAirflow	Optional	none	not permitted	Maximum airflow setpoint of a VAV terminal
SCPTminFlowHeat applying to SFPTsccVAV	Optional	none	not permitted	Minimum airflow setpoint of a VAV terminal while heating
SCPTmaxFlowHeat applying to SFPTsccVAV	Optional	none	not permitted	Maximum airflow setpoint of a VAV terminal while heating

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTminFlowStby applying to SFPTsccVAV	Optional	none	not permitted	Minimum airflow setpoint of a VAV terminal in the Standby (occupancy) mode
SCPThvacType applying to SFPTsccVAV	Optional	device specific, constant	not permitted	This value is set by the manufacturer to allow an integrator to know the function of this SCC device
SCPTfanOperation applying to SFPTsccVAV	Optional	none	not permitted	Specifies fan operation during occupied and occupied standby. Fan operation during unoccupied is manufacturer defined
SCPTminFlowUnit applying to SFPTsccVAV	Optional	none	not permitted	Unit minimum air flow for dual duct VAV Terminal units
SCPTmaxFlowUnit applying to SFPTsccVAV	Optional	none	not permitted	Unit maximum air flow for dual duct VAV Terminal units
SCPTminFlowHeatStby applying to SFPTsccVAV	Optional	none	not permitted	The heating or ventilated deck minimum flow of a dual duct VAV Terminal unit during occupied standby mode
SCPTminFlowUnitStby applying to SFPTsccVAV	Optional	none	not permitted	Total unit minimum airflow for dual duct units during occupied standby mode
SCPToffsetFlow applying to SFPTsccVAV	Optional	none	not permitted	
SCPTareaDuctHeat applying to SFPTsccVAV	Optional	none	not permitted	Nominal cross-sectional airflow area of the hot or ventilation deck of a dual duct VAV terminal unit
SCPTnomAirFlowHeat applying to SFPTsccVAV	Optional	none	not permitted	Value used to provide the nominal airflow volume of a hot or ventilation deck of a dual duct VAV terminal
SCPTgainVAVHeat applying to SFPTsccVAV	Optional	none	not permitted	Calibration constant used to calculate airflow
SCPTnumDampers applying to SFPTsccVAV	Optional	none	not permitted	Indicates to the controller if it is in a single or dual duct system

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTminFlowUnitHeat applying to SFPTsccVAV	Optional	none	not permitted	Minimum airflow setpoint of a single duct, or the unit minimum airflow setpoint of a dual duct VAV terminal when using a unit (local) heating source
SCPTsaturationDelay applying to SFPTsccVAV	Optional	none	not permitted	

7.69 SFPTsccHeatPump (8503)

NV Member	Requirement	CNP Type	Direction	Description
1 "Space temperature input" (nviSpaceTemp)	Mandatory	SNVT_temp_p	input	Connects an external space temperature sensor to the node
2 "Temperature setpoint input (absolute)" (nviSetpoint)	Optional	SNVT_temp_p	input	Allows the temperature setpoints for the occupied and standby modes to be changed via the network
3 "Setpoint offset input" (nviSetptOffset)	Optional	SNVT_temp_p	input	Shifts the effective occupied and standby temperature setpoints by adding this NV's value to the present setpoints
4 "Setpoint shift input" (nviSetptShift)	Optional	SNVT_temp_setpt	input	Shifts the effective heat/cool setpoints by adding the corresponding value in this NV to the present setpoints
5 "Occupancy scheduler input" (nviOccSchedule)	Optional	SNVT_tod_event	input	Commands the SCC into different occupancy modes
6 "Occupancy override input" (nviOccManCmd)	Optional	SNVT_occupancy	input	Commands the SCC into different occupancy modes

NV Member	Requirement	CNP Type	Direction	Description
7 "Occupancy sensor input" (nviOccSensor)	Optional	SNVT_occupancy	input	Indicates the presence of occupants in the controlled space
8 "Application mode input" (nviApplicMode)	Optional	SNVT_hvac_mode	input	Used to coordinate the SCC with any supervisory controller
9 "Heat/cool mode input" (nviHeatCool)	Optional	SNVT_hvac_mode	input	Used to coordinate the SCC with any node that may need to control the heat/cool changeover of the unit
10 "Fan speed command input" (nviFanSpeedCmd)	Optional	SNVT_switch	input	Enables connection of an external fan speed switch, or allows a supervisory device to override the fan speed
11 "Compressor enable input" (nviComprEnable)	Optional	SNVT_switch	input	This input is used to disable compressor operation
12 "Auxiliary heat enable input" (nviAuxHeatEnable)	Optional	SNVT_switch	input	This input is used to disable auxiliary heat operation
13 "Economizer enable input" (nviEconEnable)	Optional	SNVT_switch	input	This input is used to enable and disable economizer operation
14 "Energy hold-off input" (nviEnergyHoldOff)	Optional	SNVT_switch	input	This input is used to stop heating and cooling while allowing the unit to protect the space from temperature extremes
15 "Water valve override input" (nviValveOverride)	Optional	SNVT_hvac_overid	input	Commands the controller into a manual mode for overriding water valves
16 "Airflow override input" (nviFlowOverride)	Optional	SNVT_hvac_overid	input	Commands the controller into a manual mode for overriding airflow control

NV Member	Requirement	CNP Type	Direction	Description
17 "Emergency override input" (nviEmergOverride)	Optional	SNVT_hvac_emerg	input	Commands the device into different emergency modes
18 "Source temperature input" (nviSourceTemp)	Optional	SNVT_temp_p	input	Indicates the temperature of the air or water being supplied to the unit for heating and/or cooling capacity
19 "Outdoor air temperature input" (nviOutdoorTemp)	Optional	SNVT_temp_p	input	Represents information from an outdoor air temperature sensor
20 "Space humidity input" (nviSpaceRH)	Optional	SNVT_lev_percent	input	Measured space humidity in percent
21 "Outdoor air humidity input" (nviOutdoorRH)	Optional	SNVT_lev_percent	input	Measured outdoor humidity in percent
22 "Space indoor air quality input" (nviSpaceIAQ)	Optional	SNVT_ppm	input	Measured space CO2 or VOC levels in PPM
23 "Space dewpoint temperature input" (nviSpaceDewPt)	Optional	SNVT_temp_p	input	Measured space dewpoint temperature
24 "Outdoor air dewpoint temperature input" (nviOutdoorDewPt)	Optional	SNVT_temp_p	input	Measured outdoor dewpoint temperature
25 "Airflow input" (nviAirflow)	Optional	SNVT_flow	input	The measured supply airflow value is typically provided by a flow sensor on the network
26 "Effective space temperature output" (nvoSpaceTemp)	Mandatory	SNVT_temp_p	output	Used to monitor the effective space temperature that the SCC is using for control

NV Member	Requirement	CNP Type	Direction	Description
27 "Unit status output" (nvoUnitStatus)	Mandatory	SNVT_hvac_status	output	Reports the SCC status
28 "Effective setpoint output" (nvoEffectSetpt)	Optional	SNVT_temp_p	output	Monitors the effective temperature setpoint
29 "Effective occupancy output" (nvoEffectOccup)	Optional	SNVT_occupancy	output	Actual occupancy mode of the unit
30 "Effective heat/cool output" (nvoHeatCool)	Optional	SNVT_hvac_mode	output	Actual heat/cool mode of the unit
31 "Local setpoint output" (nvoSetpoint)	Optional	SNVT_temp_p	output	Space temperature setpoint value if a setpoint device is hardwired
32 "Local setpoint shift output" (nvoSetptShift)	Optional	SNVT_temp_setpt	output	Locally determined shift of the effective heat/cool setpoints
33 "Fan speed output" (nvoFanSpeed)	Optional	SNVT_switch	output	Actual fan speed of a local multi-speed fan as well as the requested speed of a remote fan
34 "Discharge air temperature output" (nvoDischAirTemp)	Optional	SNVT_temp_p	output	Monitors the temperature of the air that leaves the SCC
35 "Absolute power consumption output" (nvoLoadAbs)	Optional	SNVT_power	output	Present power consumption of the unit
36 "Absolute power consumption (kW) output" (nvoLoadAbsK)	Optional	SNVT_power_kilo	output	Present power consumption of the unit

NV Member	Requirement	CNP Type	Direction	Description
37 "Terminal load output" (nvoTerminalLoad)	Optional	SNVT_lev_percent	output	Present heat/cool energy demand of the unit
38 "Primary heat output" (nvoHeatPrimary)	Optional	SNVT_lev_percent	output	Present level of the primary heat output
39 "Secondary heat output" (nvoHeatSecondary)	Optional	SNVT_lev_percent	output	Present level of the secondary heat output
40 "Primary cool output" (nvoCoolPrimary)	Optional	SNVT_lev_percent	output	Present level of the primary mechanical cooling output
41 "Secondary cool output" (nvoCoolSecondary)	Optional	SNVT_lev_percent	output	Present level of the secondary mechanical cooling output
42 "Outdoor air damper output" (nvoOADamper)	Optional	SNVT_lev_percent	output	Present position of the outdoor air damper (if hardwired) or as a request to a remote outdoor air damper
43 "Space humidity output" (nvoSpaceRH)	Optional	SNVT_lev_percent	output	Space humidity in percent, if the SCC Device has a hardwired humidity sensor
44 "Outdoor air humidity output" (nvoOutdoorRH)	Optional	SNVT_lev_percent	output	Outdoor air humidity in percent, if the SCC Device has a hardwired humidity sensor
45 "Outdoor air temperature output" (nvoOutdoorTemp)	Optional	SNVT_temp_p	output	Monitors the outdoor air temperature if the unit controller provides a hardwired temperature sensor
46 "Space CO2 sensor output" (nvoSpaceCO2)	Optional	SNVT_ppm	output	Space CO2 concentration in ppm, if the SCC Device has a hardwired CO2 sensor

NV Member	Requirement	CNP Type	Direction	Description
47 "Space dewpoint temperature output" (nvoSpaceDewPt)	Optional	SNVT_temp_p	output	Space dewpoint temperature
48 "Humidifier output" (nvoHumidifier)	Optional	SNVT_lev_percent	output	Present value of the humidifier (if hardwired) or can be used to control a remote humidifier or control valve
49 "Energy hold-off output" (nvoEnergyHoldOff)	Optional	SNVT_switch	output	Present state of an energy hold-off device that is hardwired to the controller
50 "Effective airflow setpoint output" (nvoEffectFlowSP)	Optional	SNVT_flow	output	Active flow setpoint used by the flow control loop
51 "Flow control damper setpoint output" (nvoFlowSetpoint)	Optional	SNVT_lev_percent	output	Active flow setpoint used by the flow control loop
52 "Airflow output" (nvoAirflow)	Optional	SNVT_flow	output	Measured airflow in the unit
53 "Heat source temperature input" (nviHeatSrcTemp)	Optional	SNVT_temp_p	input	Temperature of the air or water being supplied to the unit for heating capacity
54 "Cool source temperature input" (nviCoolSrcTemp)	Optional	SNVT_temp_p	input	Temperature of the air or water being supplied to the unit for cooling capacity
55 "Primary heat input for slave operation" (nviHeatPriSlave)	Optional	SNVT_lev_percent	input	This input NV is intended for slave operation
56 "Secondary heat input for slave operation" (nviHeatSecSlave)	Optional	SNVT_lev_percent	input	This input NV is intended for slave operation

NV Member	Requirement	CNP Type	Direction	Description
57 "Primary cool input for slave operation" (nviCoolPriSlave)	Optional	SNVT_lev_percent	input	This input NV is intended for slave operation
58 "Secondary cool input for slave operation" (nviCoolSecSlave)	Optional	SNVT_lev_percent	input	This input NV is intended for slave operation
59 "Minimum position OA damper input" (nviOAMinPos)	Optional	SNVT_lev_percent	input	Dynamic minimum position setpoint for an outdoor air damper. When valid it will supersede nciOAMinPos.
60 "Minimum air flow setpoint input" (nviMinAirFlow)	Optional	SNVT_lev_percent	input	Dynamic minimum cooling airflow setpoint for single or dual duct VAV terminal units. When valid it will supersede nciMinFlow.
61 "Minimum heat air flow setpoint input" (nviMinAirFlowHt)	Optional	SNVT_lev_percent	input	Dynamic minimum heating airflow setpoint for single or dual duct VAV terminal units. When valid it will supersede nciMinFlowHeat.
62 "Air flow setpoint input" (nviAirFlowSetpt)	Optional	SNVT_flow	input	The controller will add this input with nciFlowOffset to derive the active flow setpoint
63 "Terminal load input" (nviTerminalLoad)	Optional	SNVT_lev_percent	input	When used with nvoTerminalLoad from another controller can be used to coordinate master/salve operation
64 "Mixed air temperature output" (nvoMixedAirTemp)	Optional	SNVT_temp_p	output	The temperature of the combined return and fresh airstreams in an AHU before they reach the water coils
65 "Local space temperature output" (nvoLocalSpaceTmp)	Optional	SNVT_temp_p	output	Local hardwired space temperature input
66 "Effective air flow heat setpoint output" (nvoEffFlowSPHeat)	Optional	SNVT_flow	output	The hot or ventilation duct flow setpoint of a dual duct unit.

NV Member	Requirement	CNP Type	Direction	Description
67 "Flow control damper heat setpoint output" (nvoFlowSPHeat)	Optional	SNVT_lev_percent	output	Active flow setpoint used by the flow control loop for a hot or ventilation deck in a dual duct unit
68 "Air flow heat output" (nvoAirFlowHeat)	Optional	SNVT_flow	output	Air flow of a hot or ventilation deck of a dual duct VAV terminal
69 "HVAC saturation status" (nvoSatStatus)	Optional	SNVT_hvac_satsts	output	Indicates whether the control algorithm capacity limits, or end device physical limits, have been reached

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to SFPTsccHeatPump	Mandatory	none	not permitted	Maximum period of time that expires before the specified NV outputs will automatically be updated
SCPTsetPnts applying to SFPTsccHeatPump	Mandatory	none	not permitted	Space temperature setpoints for the various heat, cool, and occupancy modes
SCPTminSendTime applying to SFPTsccHeatPump	Optional	none	not permitted	Minimum period of time between automatic NV output transmissions
SCPTmaxRcvTime applying to SFPTsccHeatPump	Optional	none	not permitted	Maximum time that elapses after the last update to a specified NV input before the SCC starts to use its default values
SCPTlocation applying to SFPTsccHeatPump	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTbypassTime applying to nviOccManCmd	Optional	none	not permitted	Maximum amount of time that the SCC can be in the bypass (occupancy) mode following a single bypass request

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmanOvrTime applying to nviValveOverride	Optional	none	not permitted	Maximum time that the SCC will stay in a manual mode that was requested by a NV input, without receiving an update on that NV
SCPTminRnge applying to nvoOADamper	Optional	none	not permitted	Minimum position for the outdoor air damper
SCPTlimitCO2 applying to nviSpaceIAQ	Optional	none	not permitted	Maximum limit to allowable carbon dioxide within a defined area
SCPThumSetpt applying to nviSpaceRH	Optional	none	not permitted	High-limit humidity setpoint for the controlled space
SCPTnumValves applying to SFPTsccHeatPump	Optional	none	not permitted	Selects whether the SCC is used in a two pipe (one valve) or four pipe (two valve) system
SCPTductArea applying to SFPTsccHeatPump	Optional	none	not permitted	Nominal cross-sectional airflow area of a VAV terminal
SCPTnomAirFlow applying to nvoFlowSetpoint	Optional	none	not permitted	Nominal airflow volume of a VAV terminal
SCPTsensConstVAV applying to SFPTsccHeatPump	Optional	none	not permitted	Calibrates the airflow reading of a VAV terminal
SCPTminFlow applying to nviAirflow	Optional	none	not permitted	Minimum airflow setpoint of a VAV terminal
SCPTmaxFlow applying to nviAirflow	Optional	none	not permitted	Maximum airflow setpoint of a VAV terminal
SCPTminFlowHeat applying to SFPTsccHeatPump	Optional	none	not permitted	Minimum airflow setpoint of a VAV terminal while heating

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxFlowHeat applying to SFPTsccHeatPump	Optional	none	not permitted	Maximum airflow setpoint of a VAV terminal while heating
SCPTminFlowStby applying to SFPTsccHeatPump	Optional	none	not permitted	Minimum airflow setpoint of a VAV terminal in the Standby (occupancy) mode
SCPThvacType applying to SFPTsccHeatPump	Optional	device specific, constant	not permitted	This value is set by the manufacturer to allow an integrator to know the function of this SCC device
SCPTfanOperation applying to SFPTsccHeatPump	Optional	none	not permitted	Specifies fan operation during occupied and occupied standby. Fan operation during unoccupied is manufacturer defined
SCPTminFlowUnit applying to SFPTsccHeatPump	Optional	none	not permitted	Unit minimum air flow for dual duct VAV Terminal units
SCPTmaxFlowUnit applying to SFPTsccHeatPump	Optional	none	not permitted	Unit maximum air flow for dual duct VAV Terminal units
SCPTminFlowHeatStby applying to SFPTsccHeatPump	Optional	none	not permitted	The heating or ventilated deck minimum flow of a dual duct VAV Terminal unit during occupied standby mode
SCPTminFlowUnitStby applying to SFPTsccHeatPump	Optional	none	not permitted	Total unit minimum airflow for dual duct units during occupied standby mode
SCPToffsetFlow applying to SFPTsccHeatPump	Optional	none	not permitted	
SCPTareaDuctHeat applying to SFPTsccHeatPump	Optional	none	not permitted	Nominal cross-sectional airflow area of the hot or ventilation deck of a dual duct VAV terminal unit

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTnomAirFlowHeat applying to SFPTsccHeatPump	Optional	none	not permitted	Value used to provide the nominal airflow volume of a hot or ventilation deck of a dual duct VAV terminal
SCPTgainVAVHeat applying to SFPTsccHeatPump	Optional	none	not permitted	Calibration constant used to calculate airflow
SCPTnumDampers applying to SFPTsccHeatPump	Optional	none	not permitted	Indicates to the controller if it is in a single or dual duct system
SCPTminFlowUnitHeat applying to SFPTsccHeatPump	Optional	none	not permitted	Minimum airflow setpoint of a single duct, or the unit minimum airflow setpoint of a dual duct VAV terminal when using a unit (local) heating source
SCPTsaturationDelay applying to SFPTsccHeatPump	Optional	none	not permitted	

7.70 SFPTsccRooftop (8504)

NV Member	Requirement	CNP Type	Direction	Description
1 "Space temperature input" (nviSpaceTemp)	Mandatory	SNVT_temp_p	input	Connects an external space temperature sensor to the node
2 "Temperature setpoint input (absolute)" (nviSetpoint)	Optional	SNVT_temp_p	input	Allows the temperature setpoints for the occupied and standby modes to be changed via the network
3 "Setpoint offset input" (nviSetptOffset)	Optional	SNVT_temp_p	input	Shifts the effective occupied and standby temperature setpoints by adding this NV's value to the present setpoints

NV Member	Requirement	CNP Type	Direction	Description
4 "Setpoint shift input" (nviSetptShift)	Optional	SNVT_temp_setpt	input	Shifts the effective heat/cool setpoints by adding the corresponding value in this NV to the present setpoints
5 "Occupancy scheduler input" (nviOccSchedule)	Optional	SNVT_tod_event	input	Commands the SCC into different occupancy modes
6 "Occupancy override input" (nviOccManCmd)	Optional	SNVT_occupancy	input	Commands the SCC into different occupancy modes
7 "Occupancy sensor input" (nviOccSensor)	Optional	SNVT_occupancy	input	Indicates the presence of occupants in the controlled space
8 "Application mode input" (nviApplicMode)	Optional	SNVT_hvac_mode	input	Used to coordinate the SCC with any supervisory controller
9 "Heat/cool mode input" (nviHeatCool)	Optional	SNVT_hvac_mode	input	Used to coordinate the SCC with any node that may need to control the heat/cool changeover of the unit
10 "Fan speed command input" (nviFanSpeedCmd)	Optional	SNVT_switch	input	Enables connection of an external fan speed switch, or allows a supervisory device to override the fan speed
11 "Compressor enable input" (nviComprEnable)	Optional	SNVT_switch	input	This input is used to disable compressor operation
12 "Auxiliary heat enable input" (nviAuxHeatEnable)	Optional	SNVT_switch	input	This input is used to disable auxiliary heat operation
13 "Economizer enable input" (nviEconEnable)	Optional	SNVT_switch	input	This input is used to enable and disable economizer operation

NV Member	Requirement	CNP Type	Direction	Description
14 "Energy hold-off input" (nviEnergyHoldOff)	Optional	SNVT_switch	input	This input is used to stop heating and cooling while allowing the unit to protect the space from temperature extremes
15 "Water valve override input" (nviValveOverride)	Optional	SNVT_hvac_overid	input	Commands the controller into a manual mode for overriding water valves
16 "Airflow override input" (nviFlowOverride)	Optional	SNVT_hvac_overid	input	Commands the controller into a manual mode for overriding airflow control
17 "Emergency override input" (nviEmergOverride)	Optional	SNVT_hvac_emerg	input	Commands the device into different emergency modes
18 "Source temperature input" (nviSourceTemp)	Optional	SNVT_temp_p	input	Indicates the temperature of the air or water being supplied to the unit for heating and/or cooling capacity
19 "Outdoor air temperature input" (nviOutdoorTemp)	Optional	SNVT_temp_p	input	Represents information from an outdoor air temperature sensor
20 "Space humidity input" (nviSpaceRH)	Optional	SNVT_lev_percent	input	Measured space humidity in percent
21 "Outdoor air humidity input" (nviOutdoorRH)	Optional	SNVT_lev_percent	input	Measured outdoor humidity in percent
22 "Space indoor air quality input" (nviSpaceIAQ)	Optional	SNVT_ppm	input	Measured space CO2 or VOC levels in PPM
23 "Space dewpoint temperature input" (nviSpaceDewPt)	Optional	SNVT_temp_p	input	Measured space dewpoint temperature

NV Member	Requirement	CNP Type	Direction	Description
24 "Outdoor air dewpoint temperature input" (nviOutdoorDewPt)	Optional	SNVT_temp_p	input	Measured outdoor dewpoint temperature
25 "Airflow input" (nviAirflow)	Optional	SNVT_flow	input	The measured supply airflow value is typically provided by a flow sensor on the network
26 "Effective space temperature output" (nvoSpaceTemp)	Mandatory	SNVT_temp_p	output	Used to monitor the effective space temperature that the SCC is using for control
27 "Unit status output" (nvoUnitStatus)	Mandatory	SNVT_hvac_status	output	Reports the SCC status
28 "Effective setpoint output" (nvoEffectSetpt)	Optional	SNVT_temp_p	output	Monitors the effective temperature setpoint
29 "Effective occupancy output" (nvoEffectOccup)	Optional	SNVT_occupancy	output	Actual occupancy mode of the unit
30 "Effective heat/cool output" (nvoHeatCool)	Optional	SNVT_hvac_mode	output	Actual heat/cool mode of the unit
31 "Local setpoint output" (nvoSetpoint)	Optional	SNVT_temp_p	output	Space temperature setpoint value if a setpoint device is hardwired
32 "Local setpoint shift output" (nvoSetptShift)	Optional	SNVT_temp_setpt	output	Locally determined shift of the effective heat/cool setpoints
33 "Fan speed output" (nvoFanSpeed)	Optional	SNVT_switch	output	Actual fan speed of a local multi-speed fan as well as the requested speed of a remote fan

NV Member	Requirement	CNP Type	Direction	Description
34 "Discharge air temperature output" (nvoDischAirTemp)	Optional	SNVT_temp_p	output	Monitors the temperature of the air that leaves the SCC
35 "Absolute power consumption output" (nvoLoadAbs)	Optional	SNVT_power	output	Present power consumption of the unit
36 "Absolute power consumption (kW) output" (nvoLoadAbsK)	Optional	SNVT_power_kilo	output	Present power consumption of the unit
37 "Terminal load output" (nvoTerminalLoad)	Optional	SNVT_lev_percent	output	Present heat/cool energy demand of the unit
38 "Primary heat output" (nvoHeatPrimary)	Optional	SNVT_lev_percent	output	Present level of the primary heat output
39 "Secondary heat output" (nvoHeatSecondary)	Optional	SNVT_lev_percent	output	Present level of the secondary heat output
40 "Primary cool output" (nvoCoolPrimary)	Optional	SNVT_lev_percent	output	Present level of the primary mechanical cooling output
41 "Secondary cool output" (nvoCoolSecondary)	Optional	SNVT_lev_percent	output	Present level of the secondary mechanical cooling output
42 "Outdoor air damper output" (nvoOADamper)	Optional	SNVT_lev_percent	output	Present position of the outdoor air damper (if hardwired) or as a request to a remote outdoor air damper
43 "Space humidity output" (nvoSpaceRH)	Optional	SNVT_lev_percent	output	Space humidity in percent, if the SCC Device has a hardwired humidity sensor

NV Member	Requirement	CNP Type	Direction	Description
44 "Outdoor air humidity output" (nvoOutdoorRH)	Optional	SNVT_lev_percent	output	Outdoor air humidity in percent, if the SCC Device has a hardwired humidity sensor
45 "Outdoor air temperature output" (nvoOutdoorTemp)	Optional	SNVT_temp_p	output	Monitors the outdoor air temperature if the unit controller provides a hardwired temperature sensor
46 "Space CO2 sensor output" (nvoSpaceCO2)	Optional	SNVT_ppm	output	Space CO2 concentration in ppm, if the SCC Device has a hardwired CO2 sensor
47 "Space dewpoint temperature output" (nvoSpaceDewPt)	Optional	SNVT_temp_p	output	Space dewpoint temperature
48 "Humidifier output" (nvoHumidifier)	Optional	SNVT_lev_percent	output	Present value of the humidifier (if hardwired) or can be used to control a remote humidifier or control valve
49 "Energy hold-off output" (nvoEnergyHoldOff)	Optional	SNVT_switch	output	Present state of an energy hold-off device that is hardwired to the controller
50 "Effective airflow setpoint output" (nvoEffectFlowSP)	Optional	SNVT_flow	output	Active flow setpoint used by the flow control loop
51 "Flow control damper setpoint output" (nvoFlowSetpoint)	Optional	SNVT_lev_percent	output	Active flow setpoint used by the flow control loop
52 "Airflow output" (nvoAirflow)	Optional	SNVT_flow	output	Measured airflow in the unit
53 "Heat source temperature input" (nviHeatSrcTemp)	Optional	SNVT_temp_p	input	Temperature of the air or water being supplied to the unit for heating capacity

NV Member	Requirement	CNP Type	Direction	Description
54 "Cool source temperature input" (nviCoolSrcTemp)	Optional	SNVT_temp_p	input	Temperature of the air or water being supplied to the unit for cooling capacity
55 "Primary heat input for slave operation" (nviHeatPriSlave)	Optional	SNVT_lev_percent	input	This input NV is intended for slave operation
56 "Secondary heat input for slave operation" (nviHeatSecSlave)	Optional	SNVT_lev_percent	input	This input NV is intended for slave operation
57 "Primary cool input for slave operation" (nviCoolPriSlave)	Optional	SNVT_lev_percent	input	This input NV is intended for slave operation
58 "Secondary cool input for slave operation" (nviCoolSecSlave)	Optional	SNVT_lev_percent	input	This input NV is intended for slave operation
59 "Minimum position OA damper input" (nviOAMinPos)	Optional	SNVT_lev_percent	input	Dynamic minimum position setpoint for an outdoor air damper. When valid it will supersede nciOAMinPos.
60 "Minimum air flow setpoint input" (nviMinAirFlow)	Optional	SNVT_lev_percent	input	Dynamic minimum cooling air flow setpoint for single or dual duct VAV terminal units. When valid it will supersede nciMinFlow.
61 "Minimum heat air flow setpoint input" (nviMinAirFlowHt)	Optional	SNVT_lev_percent	input	Dynamic minimum heating air flow setpoint for single or dual duct VAV terminal units. When valid it will supersede nciMinFlowHeat.
62 "Air flow setpoint input" (nviAirFlowSetpt)	Optional	SNVT_flow	input	The controller will add this input with nciFlowOffset to derive the active flow setpoint
63 "Terminal load input" (nviTerminalLoad)	Optional	SNVT_lev_percent	input	When used with nvoTerminalLoad from another controller can be used to coordinate master/salve operation

NV Member	Requirement	CNP Type	Direction	Description
64 "Mixed air temperature output" (nvoMixedAirTemp)	Optional	SNVT_temp_p	output	The temperature of the combined return and fresh airstreams in an AHU before they reach the water coils
65 "Local space temperature output" (nvoLocalSpaceTmp)	Optional	SNVT_temp_p	output	Local hardwired space temperature input
66 "Effective air flow heat setpoint output" (nvoEffFlowSPHeat)	Optional	SNVT_flow	output	The hot or ventilation duct flow setpoint of a dual duct unit.
67 "Flow control damper heat setpoint output" (nvoFlowSPHeat)	Optional	SNVT_lev_percent	output	Active flow setpoint used by the flow control loop for a hot or ventilation deck in a dual duct unit
68 "Air flow heat output" (nvoAirFlowHeat)	Optional	SNVT_flow	output	Air flow of a hot or ventilation deck of a dual duct VAV terminal
69 "HVAC saturation status" (nvoSatStatus)	Optional	SNVT_hvac_satsts	output	Indicates whether the control algorithm capacity limits, or end device physical limits, have been reached

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to SFPTsccRooftop	Mandatory	none	not permitted	Maximum period of time that expires before the specified NV outputs will automatically be updated
SCPTsetPnts applying to SFPTsccRooftop	Mandatory	none	not permitted	Space temperature setpoints for the various heat, cool, and occupancy modes
SCPTminSendTime applying to SFPTsccRooftop	Optional	none	not permitted	Minimum period of time between automatic NV output transmissions
SCPTmaxRcvTime applying to SFPTsccRooftop	Optional	none	not permitted	Maximum time that elapses after the last update to a specified NV input before the SCC starts to use its default values

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTlocation applying to SFPTsccRooftop	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTbypassTime applying to nviOccManCmd	Optional	none	not permitted	Maximum amount of time that the SCC can be in the bypass (occupancy) mode following a single bypass request
SCPTmanOvrTime applying to nviValveOverride	Optional	none	not permitted	Maximum time that the SCC will stay in a manual mode that was requested by a NV input, without receiving an update on that NV
SCPTminRnge applying to nvoOADamper	Optional	none	not permitted	Minimum position for the outdoor air damper
SCPTlimitCO2 applying to nviSpaceIAQ	Optional	none	not permitted	Maximum limit to allowable carbon dioxide within a defined area
SCPThumSetpt applying to nviSpaceRH	Optional	none	not permitted	High-limit humidity setpoint for the controlled space
SCPTnumValves applying to SFPTsccRooftop	Optional	none	not permitted	Selects whether the SCC is used in a two pipe (one valve) or four pipe (two valve) system
SCPTductArea applying to SFPTsccRooftop	Optional	none	not permitted	Nominal cross-sectional airflow area of a VAV terminal
SCPTnomAirFlow applying to nvoFlowSetpoint	Optional	none	not permitted	Nominal airflow volume of a VAV terminal
SCPTsensConstVAV applying to SFPTsccRooftop	Optional	none	not permitted	Calibrates the airflow reading of a VAV terminal
SCPTminFlow applying to nviAirflow	Optional	none	not permitted	Minimum airflow setpoint of a VAV terminal
SCPTmaxFlow applying to nviAirflow	Optional	none	not permitted	Maximum airflow setpoint of a VAV terminal

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTminFlowHeat applying to SFPTsccRooftop	Optional	none	not permitted	Minimum airflow setpoint of a VAV terminal while heating
SCPTmaxFlowHeat applying to SFPTsccRooftop	Optional	none	not permitted	Maximum airflow setpoint of a VAV terminal while heating
SCPTminFlowStby applying to SFPTsccRooftop	Optional	none	not permitted	Minimum airflow setpoint of a VAV terminal in the Standby (occupancy) mode
SCPThvacType applying to SFPTsccRooftop	Optional	device specific, constant	not permitted	This value is set by the manufacturer to allow an integrator to know the function of this SCC device
SCPTfanOperation applying to SFPTsccRooftop	Optional	none	not permitted	Specifies fan operation during occupied and occupied standby. Fan operation during unoccupied is manufacturer defined
SCPTminFlowUnit applying to SFPTsccRooftop	Optional	none	not permitted	Unit minimum air flow for dual duct VAV Terminal units
SCPTmaxFlowUnit applying to SFPTsccRooftop	Optional	none	not permitted	Unit maximum air flow for dual duct VAV Terminal units
SCPTminFlowHeatStby applying to SFPTsccRooftop	Optional	none	not permitted	The heating or ventilated deck minimum flow of a dual duct VAV Terminal unit during occupied standby mode
SCPTminFlowUnitStby applying to SFPTsccRooftop	Optional	none	not permitted	Total unit minimum airflow for dual duct units during occupied standby mode
SCPToffsetFlow applying to SFPTsccRooftop	Optional	none	not permitted	
SCPTareaDuctHeat applying to SFPTsccRooftop	Optional	none	not permitted	Nominal cross-sectional airflow area of the hot or ventilation deck of a dual duct VAV terminal unit
SCPTnomAirFlowHeat applying to SFPTsccRooftop	Optional	none	not permitted	Value used to provide the nominal airflow volume of a hot or ventilation deck of a dual duct VAV terminal

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTgainVAVHeat applying to SFPTsccRooftop	Optional	none	not permitted	Calibration constant used to calculate airflow
SCPTnumDampers applying to SFPTsccRooftop	Optional	none	not permitted	Indicates to the controller if it is in a single or dual duct system
SCPTminFlowUnitHeat applying to SFPTsccRooftop	Optional	none	not permitted	Minimum airflow setpoint of a single duct, or the unit minimum airflow setpoint of a dual duct VAV terminal when using a unit (local) heating source
SCPTsaturationDelay applying to SFPTsccRooftop	Optional	none	not permitted	

7.71 SFPTsccUnitVentilator (8505)

NV Member	Requirement	CNP Type	Direction	Description
1 "Space temperature input" (nviSpaceTemp)	Mandatory	SNVT_temp_p	input	Connects an external space temperature sensor to the node
2 "Temperature setpoint input (absolute)" (nviSetpoint)	Optional	SNVT_temp_p	input	Allows the temperature setpoints for the occupied and standby modes to be changed via the network
3 "Setpoint offset input" (nviSetptOffset)	Optional	SNVT_temp_p	input	Shifts the effective occupied and standby temperature setpoints by adding this NV's value to the present setpoints
4 "Setpoint shift input" (nviSetptShift)	Optional	SNVT_temp_setpt	input	Shifts the effective heat/cool setpoints by adding the corresponding value in this NV to the present setpoints
5 "Occupancy scheduler input" (nviOccSchedule)	Optional	SNVT_tod_event	input	Commands the SCC into different occupancy modes

NV Member	Requirement	CNP Type	Direction	Description
6 "Occupancy override input" (nviOccManCmd)	Optional	SNVT_occupancy	input	Commands the SCC into different occupancy modes
7 "Occupancy sensor input" (nviOccSensor)	Optional	SNVT_occupancy	input	Indicates the presence of occupants in the controlled space
8 "Application mode input" (nviApplicMode)	Optional	SNVT_hvac_mode	input	Used to coordinate the SCC with any supervisory controller
9 "Heat/cool mode input" (nviHeatCool)	Optional	SNVT_hvac_mode	input	Used to coordinate the SCC with any node that may need to control the heat/cool changeover of the unit
10 "Fan speed command input" (nviFanSpeedCmd)	Optional	SNVT_switch	input	Enables connection of an external fan speed switch, or allows a supervisory device to override the fan speed
11 "Compressor enable input" (nviComprEnable)	Optional	SNVT_switch	input	This input is used to disable compressor operation
12 "Auxiliary heat enable input" (nviAuxHeatEnable)	Optional	SNVT_switch	input	This input is used to disable auxiliary heat operation
13 "Economizer enable input" (nviEconEnable)	Optional	SNVT_switch	input	This input is used to enable and disable economizer operation
14 "Energy hold-off input" (nviEnergyHoldOff)	Optional	SNVT_switch	input	This input is used to stop heating and cooling while allowing the unit to protect the space from temperature extremes
15 "Water valve override input" (nviValveOverride)	Optional	SNVT_hvac_overid	input	Commands the controller into a manual mode for overriding water valves

NV Member	Requirement	CNP Type	Direction	Description
16 "Airflow override input" (nviFlowOverride)	Optional	SNVT_hvac_overid	input	Commands the controller into a manual mode for overriding airflow control
17 "Emergency override input" (nviEmergOverride)	Optional	SNVT_hvac_emerg	input	Commands the device into different emergency modes
18 "Source temperature input" (nviSourceTemp)	Optional	SNVT_temp_p	input	Indicates the temperature of the air or water being supplied to the unit for heating and/or cooling capacity
19 "Outdoor air temperature input" (nviOutdoorTemp)	Optional	SNVT_temp_p	input	Represents information from an outdoor air temperature sensor
20 "Space humidity input" (nviSpaceRH)	Optional	SNVT_lev_percent	input	Measured space humidity in percent
21 "Outdoor air humidity input" (nviOutdoorRH)	Optional	SNVT_lev_percent	input	Measured outdoor humidity in percent
22 "Space indoor air quality input" (nviSpaceIAQ)	Optional	SNVT_ppm	input	Measured space CO2 or VOC levels in PPM
23 "Space dewpoint temperature input" (nviSpaceDewPt)	Optional	SNVT_temp_p	input	Measured space dewpoint temperature
24 "Outdoor air dewpoint temperature input" (nviOutdoorDewPt)	Optional	SNVT_temp_p	input	Measured outdoor dewpoint temperature
25 "Airflow input" (nviAirflow)	Optional	SNVT_flow	input	The measured supply airflow value is typically provided by a flow sensor on the network

NV Member	Requirement	CNP Type	Direction	Description
26 "Effective space temperature output" (nvoSpaceTemp)	Mandatory	SNVT_temp_p	output	Used to monitor the effective space temperature that the SCC is using for control
27 "Unit status output" (nvoUnitStatus)	Mandatory	SNVT_hvac_status	output	Reports the SCC status
28 "Effective setpoint output" (nvoEffectSetpt)	Optional	SNVT_temp_p	output	Monitors the effective temperature setpoint
29 "Effective occupancy output" (nvoEffectOccup)	Optional	SNVT_occupancy	output	Actual occupancy mode of the unit
30 "Effective heat/cool output" (nvoHeatCool)	Optional	SNVT_hvac_mode	output	Actual heat/cool mode of the unit
31 "Local setpoint output" (nvoSetpoint)	Optional	SNVT_temp_p	output	Space temperature setpoint value if a setpoint device is hardwired
32 "Local setpoint shift output" (nvoSetptShift)	Optional	SNVT_temp_setpt	output	Locally determined shift of the effective heat/cool setpoints
33 "Fan speed output" (nvoFanSpeed)	Optional	SNVT_switch	output	Actual fan speed of a local multi-speed fan as well as the requested speed of a remote fan
34 "Discharge air temperature output" (nvoDischAirTemp)	Optional	SNVT_temp_p	output	Monitors the temperature of the air that leaves the SCC
35 "Absolute power consumption output" (nvoLoadAbs)	Optional	SNVT_power	output	Present power consumption of the unit

NV Member	Requirement	CNP Type	Direction	Description
36 "Absolute power consumption (kW) output" (nvoLoadAbsK)	Optional	SNVT_power_kilo	output	Present power consumption of the unit
37 "Terminal load output" (nvoTerminalLoad)	Optional	SNVT_lev_percent	output	Present heat/cool energy demand of the unit
38 "Primary heat output" (nvoHeatPrimary)	Optional	SNVT_lev_percent	output	Present level of the primary heat output
39 "Secondary heat output" (nvoHeatSecondary)	Optional	SNVT_lev_percent	output	Present level of the secondary heat output
40 "Primary cool output" (nvoCoolPrimary)	Optional	SNVT_lev_percent	output	Present level of the primary mechanical cooling output
41 "Secondary cool output" (nvoCoolSecondary)	Optional	SNVT_lev_percent	output	Present level of the secondary mechanical cooling output
42 "Outdoor air damper output" (nvoOADamper)	Optional	SNVT_lev_percent	output	Present position of the outdoor air damper (if hardwired) or as a request to a remote outdoor air damper
43 "Space humidity output" (nvoSpaceRH)	Optional	SNVT_lev_percent	output	Space humidity in percent, if the SCC Device has a hardwired humidity sensor
44 "Outdoor air humidity output" (nvoOutdoorRH)	Optional	SNVT_lev_percent	output	Outdoor air humidity in percent, if the SCC Device has a hardwired humidity sensor
45 "Outdoor air temperature output" (nvoOutdoorTemp)	Optional	SNVT_temp_p	output	Monitors the outdoor air temperature if the unit controller provides a hardwired temperature sensor

NV Member	Requirement	CNP Type	Direction	Description
46 "Space CO2 sensor output" (nvoSpaceCO2)	Optional	SNVT_ppm	output	Space CO2 concentration in ppm, if the SCC Device has a hardwired CO2 sensor
47 "Space dewpoint temperature output" (nvoSpaceDewPt)	Optional	SNVT_temp_p	output	Space dewpoint temperature
48 "Humidifier output" (nvoHumidifier)	Optional	SNVT_lev_percent	output	Present value of the humidifier (if hardwired) or can be used to control a remote humidifier or control valve
49 "Energy hold-off output" (nvoEnergyHoldOff)	Optional	SNVT_switch	output	Present state of an energy hold-off device that is hardwired to the controller
50 "Effective airflow setpoint output" (nvoEffectFlowSP)	Optional	SNVT_flow	output	Active flow setpoint used by the flow control loop
51 "Flow control damper setpoint output" (nvoFlowSetpoint)	Optional	SNVT_lev_percent	output	Active flow setpoint used by the flow control loop
52 "Airflow output" (nvoAirflow)	Optional	SNVT_flow	output	Measured airflow in the unit
53 "Heat source temperature input" (nviHeatSrcTemp)	Optional	SNVT_temp_p	input	Temperature of the air or water being supplied to the unit for heating capacity
54 "Cool source temperature input" (nviCoolSrcTemp)	Optional	SNVT_temp_p	input	Temperature of the air or water being supplied to the unit for cooling capacity
55 "Primary heat input for slave operation" (nviHeatPriSlave)	Optional	SNVT_lev_percent	input	This input NV is intended for slave operation

NV Member	Requirement	CNP Type	Direction	Description
56 "Secondary heat input for slave operation" (nviHeatSecSlave)	Optional	SNVT_lev_percent	input	This input NV is intended for slave operation
57 "Primary cool input for slave operation" (nviCoolPriSlave)	Optional	SNVT_lev_percent	input	This input NV is intended for slave operation
58 "Secondary cool input for slave operation" (nviCoolSecSlave)	Optional	SNVT_lev_percent	input	This input NV is intended for slave operation
59 "Minimum position OA damper input" (nviOAMinPos)	Optional	SNVT_lev_percent	input	Dynamic minimum position setpoint for an outdoor air damper. When valid it will supersede nciOAMinPos.
60 "Minimum air flow setpoint input" (nviMinAirFlow)	Optional	SNVT_lev_percent	input	Dynamic minimum cooling air flow setpoint for single or dual duct VAV terminal units. When valid it will supersede nciMinFlow.
61 "Minimum heat air flow setpoint input" (nviMinAirFlowHt)	Optional	SNVT_lev_percent	input	Dynamic minimum heating air flow setpoint for single or dual duct VAV terminal units. When valid it will supersede nciMinFlowHeat.
62 "Air flow setpoint input" (nviAirFlowSetpt)	Optional	SNVT_flow	input	The controller will add this input with nciFlowOffset to derive the active flow setpoint
63 "Terminal load input" (nviTerminalLoad)	Optional	SNVT_lev_percent	input	When used with nvoTerminalLoad from another controller can be used to coordinate master/salve operation
64 "Mixed air temperature output" (nvoMixedAirTemp)	Optional	SNVT_temp_p	output	The temperature of the combined return and fresh airstreams in an AHU before they reach the water coils
65 "Local space temperature output" (nvoLocalSpaceTmp)	Optional	SNVT_temp_p	output	Local hardwired space temperature input

NV Member	Requirement	CNP Type	Direction	Description
66 "Effective air flow heat setpoint output" (nvoEffFlowSPHeat)	Optional	SNVT_flow	output	The hot or ventilation duct flow setpoint of a dual duct unit.
67 "Flow control damper heat setpoint output" (nvoFlowSPHeat)	Optional	SNVT_lev_percent	output	Active flow setpoint used by the flow control loop for a hot or ventilation deck in a dual duct unit
68 "Air flow heat output" (nvoAirFlowHeat)	Optional	SNVT_flow	output	Air flow of a hot or ventilation deck of a dual duct VAV terminal
69 "HVAC saturation status" (nvoSatStatus)	Optional	SNVT_hvac_satsts	output	Indicates whether the control algorithm capacity limits, or end device physical limits, have been reached

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to SFPTsccUnitVentilator	Mandatory	none	not permitted	Maximum period of time that expires before the specified NV outputs will automatically be updated
SCPTsetPnts applying to SFPTsccUnitVentilator	Mandatory	none	not permitted	Space temperature setpoints for the various heat, cool, and occupancy modes
SCPTminSendTime applying to SFPTsccUnitVentilator	Optional	none	not permitted	Minimum period of time between automatic NV output transmissions
SCPTmaxRcvTime applying to SFPTsccUnitVentilator	Optional	none	not permitted	Maximum time that elapses after the last update to a specified NV input before the SCC starts to use its default values
SCPTlocation applying to SFPTsccUnitVentilator	Optional	none	not permitted	Provides descriptive physical location information related to the object.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTbypassTime applying to nviOccManCmd	Optional	none	not permitted	Maximum amount of time that the SCC can be in the bypass (occupancy) mode following a single bypass request
SCPTmanOvrTime applying to nviValveOverride	Optional	none	not permitted	Maximum time that the SCC will stay in a manual mode that was requested by a NV input, without receiving an update on that NV
SCPTminRnge applying to nvoOADamper	Optional	none	not permitted	Minimum position for the outdoor air damper
SCPTlimitCO2 applying to nviSpaceIAQ	Optional	none	not permitted	Maximum limit to allowable carbon dioxide within a defined area
SCPThumSetpt applying to nviSpaceRH	Optional	none	not permitted	High-limit humidity setpoint for the controlled space
SCPTnumValves applying to SFPTsccUnitVentilator	Optional	none	not permitted	Selects whether the SCC is used in a two pipe (one valve) or four pipe (two valve) system
SCPTductArea applying to SFPTsccUnitVentilator	Optional	none	not permitted	Nominal cross-sectional airflow area of a VAV terminal
SCPTnomAirFlow applying to nvoFlowSetpoint	Optional	none	not permitted	Nominal airflow volume of a VAV terminal
SCPTsensConstVAV applying to SFPTsccUnitVentilator	Optional	none	not permitted	Calibrates the airflow reading of a VAV terminal
SCPTminFlow applying to nviAirflow	Optional	none	not permitted	Minimum airflow setpoint of a VAV terminal
SCPTmaxFlow applying to nviAirflow	Optional	none	not permitted	Maximum airflow setpoint of a VAV terminal

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTminFlowHeat applying to SFPTsccUnitVentilator	Optional	none	not permitted	Minimum airflow setpoint of a VAV terminal while heating
SCPTmaxFlowHeat applying to SFPTsccUnitVentilator	Optional	none	not permitted	Maximum airflow setpoint of a VAV terminal while heating
SCPTminFlowStby applying to SFPTsccUnitVentilator	Optional	none	not permitted	Minimum airflow setpoint of a VAV terminal in the Standby (occupancy) mode
SCPThvacType applying to SFPTsccUnitVentilator	Optional	device specific, constant	not permitted	This value is set by the manufacturer to allow an integrator to know the function of this SCC device
SCPTfanOperation applying to SFPTsccUnitVentilator	Optional	none	not permitted	Specifies fan operation during occupied and occupied standby. Fan operation during unoccupied is manufacturer defined
SCPTminFlowUnit applying to SFPTsccUnitVentilator	Optional	none	not permitted	Unit minimum air flow for dual duct VAV Terminal units
SCPTmaxFlowUnit applying to SFPTsccUnitVentilator	Optional	none	not permitted	Unit maximum air flow for dual duct VAV Terminal units
SCPTminFlowHeatStby applying to SFPTsccUnitVentilator	Optional	none	not permitted	The heating or ventilated deck minimum flow of a dual duct VAV Terminal unit during occupied standby mode
SCPTminFlowUnitStby applying to SFPTsccUnitVentilator	Optional	none	not permitted	Total unit minimum airflow for dual duct units during occupied standby mode
SCPToffsetFlow applying to SFPTsccUnitVentilator	Optional	none	not permitted	

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTareaDuctHeat applying to SFPTsccUnitVentilator	Optional	none	not permitted	Nominal cross-sectional airflow area of the hot or ventilation deck of a dual duct VAV terminal unit
SCPTnomAirFlowHeat applying to SFPTsccUnitVentilator	Optional	none	not permitted	Value used to provide the nominal airflow volume of a hot or ventilation deck of a dual duct VAV terminal
SCPTgainVAVHeat applying to SFPTsccUnitVentilator	Optional	none	not permitted	Calibration constant used to calculate airflow
SCPTnumDampers applying to SFPTsccUnitVentilator	Optional	none	not permitted	Indicates to the controller if it is in a single or dual duct system
SCPTminFlowUnitHeat applying to SFPTsccUnitVentilator	Optional	none	not permitted	Minimum airflow setpoint of a single duct, or the unit minimum airflow setpoint of a dual duct VAV terminal when using a unit (local) heating source
SCPTsaturationDelay applying to SFPTsccUnitVentilator	Optional	none	not permitted	

7.72 SFPTsccChilledCeiling (8506)

NV Member	Requirement	CNP Type	Direction	Description
1 "Space temperature input" (nviSpaceTemp)	Mandatory	SNVT_temp_p	input	Connects an external space temperature sensor to the node
2 "Temperature setpoint input (absolute)" (nviSetpoint)	Optional	SNVT_temp_p	input	Allows the temperature setpoints for the occupied and standby modes to be changed via the network

NV Member	Requirement	CNP Type	Direction	Description
3 "Setpoint offset input" (nviSetptOffset)	Optional	SNVT_temp_p	input	Shifts the effective occupied and standby temperature setpoints by adding this NV's value to the present setpoints
4 "Setpoint shift input" (nviSetptShift)	Optional	SNVT_temp_setpt	input	Shifts the effective heat/cool setpoints by adding the corresponding value in this NV to the present setpoints
5 "Occupancy scheduler input" (nviOccSchedule)	Optional	SNVT_tod_event	input	Commands the SCC into different occupancy modes
6 "Occupancy override input" (nviOccManCmd)	Optional	SNVT_occupancy	input	Commands the SCC into different occupancy modes
7 "Occupancy sensor input" (nviOccSensor)	Optional	SNVT_occupancy	input	Indicates the presence of occupants in the controlled space
8 "Application mode input" (nviApplicMode)	Optional	SNVT_hvac_mode	input	Used to coordinate the SCC with any supervisory controller
9 "Heat/cool mode input" (nviHeatCool)	Optional	SNVT_hvac_mode	input	Used to coordinate the SCC with any node that may need to control the heat/cool changeover of the unit
10 "Fan speed command input" (nviFanSpeedCmd)	Optional	SNVT_switch	input	Enables connection of an external fan speed switch, or allows a supervisory device to override the fan speed
11 "Compressor enable input" (nviComprEnable)	Optional	SNVT_switch	input	This input is used to disable compressor operation
12 "Auxiliary heat enable input" (nviAuxHeatEnable)	Optional	SNVT_switch	input	This input is used to disable auxiliary heat operation

NV Member	Requirement	CNP Type	Direction	Description
13 "Economizer enable input" (nviEconEnable)	Optional	SNVT_switch	input	This input is used to enable and disable economizer operation
14 "Energy hold-off input" (nviEnergyHoldOff)	Optional	SNVT_switch	input	This input is used to stop heating and cooling while allowing the unit to protect the space from temperature extremes
15 "Water valve override input" (nviValveOverride)	Optional	SNVT_hvac_overid	input	Commands the controller into a manual mode for overriding water valves
16 "Airflow override input" (nviFlowOverride)	Optional	SNVT_hvac_overid	input	Commands the controller into a manual mode for overriding airflow control
17 "Emergency override input" (nviEmergOverride)	Optional	SNVT_hvac_emerg	input	Commands the device into different emergency modes
18 "Source temperature input" (nviSourceTemp)	Optional	SNVT_temp_p	input	Indicates the temperature of the air or water being supplied to the unit for heating and/or cooling capacity
19 "Outdoor air temperature input" (nviOutdoorTemp)	Optional	SNVT_temp_p	input	Represents information from an outdoor air temperature sensor
20 "Space humidity input" (nviSpaceRH)	Optional	SNVT_lev_percent	input	Measured space humidity in percent
21 "Outdoor air humidity input" (nviOutdoorRH)	Optional	SNVT_lev_percent	input	Measured outdoor humidity in percent
22 "Space indoor air quality input" (nviSpaceIAQ)	Optional	SNVT_ppm	input	Measured space CO2 or VOC levels in PPM

NV Member	Requirement	CNP Type	Direction	Description
23 "Space dewpoint temperature input" (nviSpaceDewPt)	Optional	SNVT_temp_p	input	Measured space dewpoint temperature
24 "Outdoor air dewpoint temperature input" (nviOutdoorDewPt)	Optional	SNVT_temp_p	input	Measured outdoor dewpoint temperature
25 "Airflow input" (nviAirflow)	Optional	SNVT_flow	input	The measured supply airflow value is typically provided by a flow sensor on the network
26 "Effective space temperature output" (nvoSpaceTemp)	Mandatory	SNVT_temp_p	output	Used to monitor the effective space temperature that the SCC is using for control
27 "Unit status output" (nvoUnitStatus)	Mandatory	SNVT_hvac_status	output	Reports the SCC status
28 "Effective setpoint output" (nvoEffectSetpt)	Optional	SNVT_temp_p	output	Monitors the effective temperature setpoint
29 "Effective occupancy output" (nvoEffectOccup)	Optional	SNVT_occupancy	output	Actual occupancy mode of the unit
30 "Effective heat/cool output" (nvoHeatCool)	Optional	SNVT_hvac_mode	output	Actual heat/cool mode of the unit
31 "Local setpoint output" (nvoSetpoint)	Optional	SNVT_temp_p	output	Space temperature setpoint value if a setpoint device is hardwired
32 "Local setpoint shift output" (nvoSetptShift)	Optional	SNVT_temp_setpt	output	Locally determined shift of the effective heat/cool setpoints

NV Member	Requirement	CNP Type	Direction	Description
33 "Fan speed output" (nvoFanSpeed)	Optional	SNVT_switch	output	Actual fan speed of a local multi-speed fan as well as the requested speed of a remote fan
34 "Discharge air temperature output" (nvoDischAirTemp)	Optional	SNVT_temp_p	output	Monitors the temperature of the air that leaves the SCC
35 "Absolute power consumption output" (nvoLoadAbs)	Optional	SNVT_power	output	Present power consumption of the unit
36 "Absolute power consumption (kW) output" (nvoLoadAbsK)	Optional	SNVT_power_kilo	output	Present power consumption of the unit
37 "Terminal load output" (nvoTerminalLoad)	Optional	SNVT_lev_percent	output	Present heat/cool energy demand of the unit
38 "Primary heat output" (nvoHeatPrimary)	Optional	SNVT_lev_percent	output	Present level of the primary heat output
39 "Secondary heat output" (nvoHeatSecondary)	Optional	SNVT_lev_percent	output	Present level of the secondary heat output
40 "Primary cool output" (nvoCoolPrimary)	Optional	SNVT_lev_percent	output	Present level of the primary mechanical cooling output
41 "Secondary cool output" (nvoCoolSecondary)	Optional	SNVT_lev_percent	output	Present level of the secondary mechanical cooling output
42 "Outdoor air damper output" (nvoOADamper)	Optional	SNVT_lev_percent	output	Present position of the outdoor air damper (if hardwired) or as a request to a remote outdoor air damper

NV Member	Requirement	CNP Type	Direction	Description
43 "Space humidity output" (nvoSpaceRH)	Optional	SNVT_lev_percent	output	Space humidity in percent, if the SCC Device has a hardwired humidity sensor
44 "Outdoor air humidity output" (nvoOutdoorRH)	Optional	SNVT_lev_percent	output	Outdoor air humidity in percent, if the SCC Device has a hardwired humidity sensor
45 "Outdoor air temperature output" (nvoOutdoorTemp)	Optional	SNVT_temp_p	output	Monitors the outdoor air temperature if the unit controller provides a hardwired temperature sensor
46 "Space CO2 sensor output" (nvoSpaceCO2)	Optional	SNVT_ppm	output	Space CO2 concentration in ppm, if the SCC Device has a hardwired CO2 sensor
47 "Space dewpoint temperature output" (nvoSpaceDewPt)	Optional	SNVT_temp_p	output	Space dewpoint temperature
48 "Humidifier output" (nvoHumidifier)	Optional	SNVT_lev_percent	output	Present value of the humidifier (if hardwired) or can be used to control a remote humidifier or control valve
49 "Energy hold-off output" (nvoEnergyHoldOff)	Optional	SNVT_switch	output	Present state of an energy hold-off device that is hardwired to the controller
50 "Effective airflow setpoint output" (nvoEffectFlowSP)	Optional	SNVT_flow	output	Active flow setpoint used by the flow control loop
51 "Flow control damper setpoint output" (nvoFlowSetpoint)	Optional	SNVT_lev_percent	output	Active flow setpoint used by the flow control loop
52 "Airflow output" (nvoAirflow)	Optional	SNVT_flow	output	Measured airflow in the unit

NV Member	Requirement	CNP Type	Direction	Description
53 "Heat source temperature input" (nviHeatSrcTemp)	Optional	SNVT_temp_p	input	Temperature of the air or water being supplied to the unit for heating capacity
54 "Cool source temperature input" (nviCoolSrcTemp)	Optional	SNVT_temp_p	input	Temperature of the air or water being supplied to the unit for cooling capacity
55 "Primary heat input for slave operation" (nviHeatPriSlave)	Optional	SNVT_lev_percent	input	This input NV is intended for slave operation
56 "Secondary heat input for slave operation" (nviHeatSecSlave)	Optional	SNVT_lev_percent	input	This input NV is intended for slave operation
57 "Primary cool input for slave operation" (nviCoolPriSlave)	Optional	SNVT_lev_percent	input	This input NV is intended for slave operation
58 "Secondary cool input for slave operation" (nviCoolSecSlave)	Optional	SNVT_lev_percent	input	This input NV is intended for slave operation
59 "Minimum position OA damper input" (nviOAMinPos)	Optional	SNVT_lev_percent	input	Dynamic minimum position setpoint for an outdoor air damper. When valid it will supersede nciOAMinPos.
60 "Minimum air flow setpoint input" (nviMinAirFlow)	Optional	SNVT_lev_percent	input	Dynamic minimum cooling air flow setpoint for single or dual duct VAV terminal units. When valid it will supersede nciMinFlow.
61 "Minimum heat air flow setpoint input" (nviMinAirFlowHt)	Optional	SNVT_lev_percent	input	Dynamic minimum heating air flow setpoint for single or dual duct VAV terminal units. When valid it will supersede nciMinFlowHeat.
62 "Air flow setpoint input" (nviAirFlowSetpt)	Optional	SNVT_flow	input	The controller will add this input with nciFlowOffset to derive the active flow setpoint

NV Member	Requirement	CNP Type	Direction	Description
63 "Terminal load input" (nviTerminalLoad)	Optional	SNVT_lev_percent	input	When used with nvoTerminalLoad from another controller can be used to coordinate master/salve operation
64 "Mixed air temperature output" (nvoMixedAirTemp)	Optional	SNVT_temp_p	output	The temperature of the combined return and fresh airstreams in an AHU before they reach the water coils
65 "Local space temperature output" (nvoLocalSpaceTmp)	Optional	SNVT_temp_p	output	Local hardwired space temperature input
66 "Effective air flow heat setpoint output" (nvoEffFlowSPHeat)	Optional	SNVT_flow	output	The hot or ventilation duct flow setpoint of a dual duct unit.
67 "Flow control damper heat setpoint output" (nvoFlowSPHeat)	Optional	SNVT_lev_percent	output	Active flow setpoint used by the flow control loop for a hot or ventilation deck in a dual duct unit
68 "Air flow heat output" (nvoAirFlowHeat)	Optional	SNVT_flow	output	Air flow of a hot or ventilation deck of a dual duct VAV terminal
69 "HVAC saturation status" (nvoSatStatus)	Optional	SNVT_hvac_satsts	output	Indicates whether the control algorithm capacity limits, or end device physical limits, have been reached

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to SFPTsccChilledCeiling	Mandatory	none	not permitted	Maximum period of time that expires before the specified NV outputs will automatically be updated
SCPTsetPnts applying to SFPTsccChilledCeiling	Mandatory	none	not permitted	Space temperature setpoints for the various heat, cool, and occupancy modes

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTminSendTime applying to SFPTsccChilledCeiling	Optional	none	not permitted	Minimum period of time between automatic NV output transmissions
SCPTmaxRcvTime applying to SFPTsccChilledCeiling	Optional	none	not permitted	Maximum time that elapses after the last update to a specified NV input before the SCC starts to use its default values
SCPTlocation applying to SFPTsccChilledCeiling	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTbypassTime applying to nviOccManCmd	Optional	none	not permitted	Maximum amount of time that the SCC can be in the bypass (occupancy) mode following a single bypass request
SCPTmanOvrTime applying to nviValveOverride	Optional	none	not permitted	Maximum time that the SCC will stay in a manual mode that was requested by a NV input, without receiving an update on that NV
SCPTminRnge applying to nvoOADamper	Optional	none	not permitted	Minimum position for the outdoor air damper
SCPTlimitCO2 applying to nviSpaceIAQ	Optional	none	not permitted	Maximum limit to allowable carbon dioxide within a defined area
SCPThumSetpt applying to nviSpaceRH	Optional	none	not permitted	High-limit humidity setpoint for the controlled space
SCPTnumValves applying to SFPTsccChilledCeiling	Optional	none	not permitted	Selects whether the SCC is used in a two pipe (one valve) or four pipe (two valve) system
SCPTductArea applying to SFPTsccChilledCeiling	Optional	none	not permitted	Nominal cross-sectional airflow area of a VAV terminal
SCPTnomAirFlow applying to nvoFlowSetpoint	Optional	none	not permitted	Nominal airflow volume of a VAV terminal

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTsensConstVAV applying to SFPTsccChilledCeiling	Optional	none	not permitted	Calibrates the airflow reading of a VAV terminal
SCPTminFlow applying to nviAirflow	Optional	none	not permitted	Minimum airflow setpoint of a VAV terminal
SCPTmaxFlow applying to nviAirflow	Optional	none	not permitted	Maximum airflow setpoint of a VAV terminal
SCPTminFlowHeat applying to SFPTsccChilledCeiling	Optional	none	not permitted	Minimum airflow setpoint of a VAV terminal while heating
SCPTmaxFlowHeat applying to SFPTsccChilledCeiling	Optional	none	not permitted	Maximum airflow setpoint of a VAV terminal while heating
SCPTminFlowStby applying to SFPTsccChilledCeiling	Optional	none	not permitted	Minimum airflow setpoint of a VAV terminal in the Standby (occupancy) mode
SCPThvacType applying to SFPTsccChilledCeiling	Optional	device specific, constant	not permitted	This value is set by the manufacturer to allow an integrator to know the function of this SCC device
SCPTfanOperation applying to SFPTsccChilledCeiling	Optional	none	not permitted	Specifies fan operation during occupied and occupied standby. Fan operation during unoccupied is manufacturer defined
SCPTminFlowUnit applying to SFPTsccChilledCeiling	Optional	none	not permitted	Unit minimum air flow for dual duct VAV Terminal units
SCPTmaxFlowUnit applying to SFPTsccChilledCeiling	Optional	none	not permitted	Unit maximum air flow for dual duct VAV Terminal units

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTminFlowHeatStby applying to SFPTsccChilledCeiling	Optional	none	not permitted	The heating or ventilated deck minimum flow of a dual duct VAV Terminal unit during occupied standby mode
SCPTminFlowUnitStby applying to SFPTsccChilledCeiling	Optional	none	not permitted	Total unit minimum airflow for dual duct units during occupied standby mode
SCPToffsetFlow applying to SFPTsccChilledCeiling	Optional	none	not permitted	
SCPTareaDuctHeat applying to SFPTsccChilledCeiling	Optional	none	not permitted	Nominal cross-sectional airflow area of the hot or ventilation deck of a dual duct VAV terminal unit
SCPTnomAirFlowHeat applying to SFPTsccChilledCeiling	Optional	none	not permitted	Value used to provide the nominal airflow volume of a hot or ventilation deck of a dual duct VAV terminal
SCPTgainVAVHeat applying to SFPTsccChilledCeiling	Optional	none	not permitted	Calibration constant used to calculate airflow
SCPTnumDampers applying to SFPTsccChilledCeiling	Optional	none	not permitted	Indicates to the controller if it is in a single or dual duct system
SCPTminFlowUnitHeat applying to SFPTsccChilledCeiling	Optional	none	not permitted	Minimum airflow setpoint of a single duct, or the unit minimum airflow setpoint of a dual duct VAV terminal when using a unit (local) heating source
SCPTsaturationDelay applying to SFPTsccChilledCeiling	Optional	none	not permitted	

7.73 SFPTsccRadiator (8507)

NV Member	Requirement	CNP Type	Direction	Description
1 "Space temperature input" (nviSpaceTemp)	Mandatory	SNVT_temp_p	input	Connects an external space temperature sensor to the node
2 "Temperature setpoint input (absolute)" (nviSetpoint)	Optional	SNVT_temp_p	input	Allows the temperature setpoints for the occupied and standby modes to be changed via the network
3 "Setpoint offset input" (nviSetptOffset)	Optional	SNVT_temp_p	input	Shifts the effective occupied and standby temperature setpoints by adding this NV's value to the present setpoints
4 "Setpoint shift input" (nviSetptShift)	Optional	SNVT_temp_setpt	input	Shifts the effective heat/cool setpoints by adding the corresponding value in this NV to the present setpoints
5 "Occupancy scheduler input" (nviOccSchedule)	Optional	SNVT_tod_event	input	Commands the SCC into different occupancy modes
6 "Occupancy override input" (nviOccManCmd)	Optional	SNVT_occupancy	input	Commands the SCC into different occupancy modes
7 "Occupancy sensor input" (nviOccSensor)	Optional	SNVT_occupancy	input	Indicates the presence of occupants in the controlled space
8 "Application mode input" (nviApplicMode)	Optional	SNVT_hvac_mode	input	Used to coordinate the SCC with any supervisory controller
9 "Heat/cool mode input" (nviHeatCool)	Optional	SNVT_hvac_mode	input	Used to coordinate the SCC with any node that may need to control the heat/cool changeover of the unit

NV Member	Requirement	CNP Type	Direction	Description
10 "Fan speed command input" (nviFanSpeedCmd)	Optional	SNVT_switch	input	Enables connection of an external fan speed switch, or allows a supervisory device to override the fan speed
11 "Compressor enable input" (nviComprEnable)	Optional	SNVT_switch	input	This input is used to disable compressor operation
12 "Auxiliary heat enable input" (nviAuxHeatEnable)	Optional	SNVT_switch	input	This input is used to disable auxiliary heat operation
13 "Economizer enable input" (nviEconEnable)	Optional	SNVT_switch	input	This input is used to enable and disable economizer operation
14 "Energy hold-off input" (nviEnergyHoldOff)	Optional	SNVT_switch	input	This input is used to stop heating and cooling while allowing the unit to protect the space from temperature extremes
15 "Water valve override input" (nviValveOverride)	Optional	SNVT_hvac_overid	input	Commands the controller into a manual mode for overriding water valves
16 "Airflow override input" (nviFlowOverride)	Optional	SNVT_hvac_overid	input	Commands the controller into a manual mode for overriding airflow control
17 "Emergency override input" (nviEmergOverride)	Optional	SNVT_hvac_emerg	input	Commands the device into different emergency modes
18 "Source temperature input" (nviSourceTemp)	Optional	SNVT_temp_p	input	Indicates the temperature of the air or water being supplied to the unit for heating and/or cooling capacity
19 "Outdoor air temperature input" (nviOutdoorTemp)	Optional	SNVT_temp_p	input	Represents information from an outdoor air temperature sensor

NV Member	Requirement	CNP Type	Direction	Description
20 "Space humidity input" (nviSpaceRH)	Optional	SNVT_lev_percent	input	Measured space humidity in percent
21 "Outdoor air humidity input" (nviOutdoorRH)	Optional	SNVT_lev_percent	input	Measured outdoor humidity in percent
22 "Space indoor air quality input" (nviSpaceIAQ)	Optional	SNVT_ppm	input	Measured space CO2 or VOC levels in PPM
23 "Space dewpoint temperature input" (nviSpaceDewPt)	Optional	SNVT_temp_p	input	Measured space dewpoint temperature
24 "Outdoor air dewpoint temperature input" (nviOutdoorDewPt)	Optional	SNVT_temp_p	input	Measured outdoor dewpoint temperature
25 "Airflow input" (nviAirflow)	Optional	SNVT_flow	input	The measured supply airflow value is typically provided by a flow sensor on the network
26 "Effective space temperature output" (nvoSpaceTemp)	Mandatory	SNVT_temp_p	output	Used to monitor the effective space temperature that the SCC is using for control
27 "Unit status output" (nvoUnitStatus)	Mandatory	SNVT_hvac_status	output	Reports the SCC status
28 "Effective setpoint output" (nvoEffectSetpt)	Optional	SNVT_temp_p	output	Monitors the effective temperature setpoint
29 "Effective occupancy output" (nvoEffectOccup)	Optional	SNVT_occupancy	output	Actual occupancy mode of the unit

NV Member	Requirement	CNP Type	Direction	Description
30 "Effective heat/cool output" (nvoHeatCool)	Optional	SNVT_hvac_mode	output	Actual heat/cool mode of the unit
31 "Local setpoint output" (nvoSetpoint)	Optional	SNVT_temp_p	output	Space temperature setpoint value if a setpoint device is hardwired
32 "Local setpoint shift output" (nvoSetptShift)	Optional	SNVT_temp_setpt	output	Locally determined shift of the effective heat/cool setpoints
33 "Fan speed output" (nvoFanSpeed)	Optional	SNVT_switch	output	Actual fan speed of a local multi-speed fan as well as the requested speed of a remote fan
34 "Discharge air temperature output" (nvoDischAirTemp)	Optional	SNVT_temp_p	output	Monitors the temperature of the air that leaves the SCC
35 "Absolute power consumption output" (nvoLoadAbs)	Optional	SNVT_power	output	Present power consumption of the unit
36 "Absolute power consumption (kW) output" (nvoLoadAbsK)	Optional	SNVT_power_kilo	output	Present power consumption of the unit
37 "Terminal load output" (nvoTerminalLoad)	Optional	SNVT_lev_percent	output	Present heat/cool energy demand of the unit
38 "Primary heat output" (nvoHeatPrimary)	Optional	SNVT_lev_percent	output	Present level of the primary heat output
39 "Secondary heat output" (nvoHeatSecondary)	Optional	SNVT_lev_percent	output	Present level of the secondary heat output

NV Member	Requirement	CNP Type	Direction	Description
40 "Primary cool output" (nvoCoolPrimary)	Optional	SNVT_lev_percent	output	Present level of the primary mechanical cooling output
41 "Secondary cool output" (nvoCoolSecondary)	Optional	SNVT_lev_percent	output	Present level of the secondary mechanical cooling output
42 "Outdoor air damper output" (nvoOADamper)	Optional	SNVT_lev_percent	output	Present position of the outdoor air damper (if hardwired) or as a request to a remote outdoor air damper
43 "Space humidity output" (nvoSpaceRH)	Optional	SNVT_lev_percent	output	Space humidity in percent, if the SCC Device has a hardwired humidity sensor
44 "Outdoor air humidity output" (nvoOutdoorRH)	Optional	SNVT_lev_percent	output	Outdoor air humidity in percent, if the SCC Device has a hardwired humidity sensor
45 "Outdoor air temperature output" (nvoOutdoorTemp)	Optional	SNVT_temp_p	output	Monitors the outdoor air temperature if the unit controller provides a hardwired temperature sensor
46 "Space CO2 sensor output" (nvoSpaceCO2)	Optional	SNVT_ppm	output	Space CO2 concentration in ppm, if the SCC Device has a hardwired CO2 sensor
47 "Space dewpoint temperature output" (nvoSpaceDewPt)	Optional	SNVT_temp_p	output	Space dewpoint temperature
48 "Humidifier output" (nvoHumidifier)	Optional	SNVT_lev_percent	output	Present value of the humidifier (if hardwired) or can be used to control a remote humidifier or control valve
49 "Energy hold-off output" (nvoEnergyHoldOff)	Optional	SNVT_switch	output	Present state of an energy hold-off device that is hardwired to the controller

NV Member	Requirement	CNP Type	Direction	Description
50 "Effective airflow setpoint output" (nvoEffectFlowSP)	Optional	SNVT_flow	output	Active flow setpoint used by the flow control loop
51 "Flow control damper setpoint output" (nvoFlowSetpoint)	Optional	SNVT_lev_percent	output	Active flow setpoint used by the flow control loop
52 "Airflow output" (nvoAirflow)	Optional	SNVT_flow	output	Measured airflow in the unit
53 "Heat source temperature input" (nviHeatSrcTemp)	Optional	SNVT_temp_p	input	Temperature of the air or water being supplied to the unit for heating capacity
54 "Cool source temperature input" (nviCoolSrcTemp)	Optional	SNVT_temp_p	input	Temperature of the air or water being supplied to the unit for cooling capacity
55 "Primary heat input for slave operation" (nviHeatPriSlave)	Optional	SNVT_lev_percent	input	This input NV is intended for slave operation
56 "Secondary heat input for slave operation" (nviHeatSecSlave)	Optional	SNVT_lev_percent	input	This input NV is intended for slave operation
57 "Primary cool input for slave operation" (nviCoolPriSlave)	Optional	SNVT_lev_percent	input	This input NV is intended for slave operation
58 "Secondary cool input for slave operation" (nviCoolSecSlave)	Optional	SNVT_lev_percent	input	This input NV is intended for slave operation
59 "Minimum position OA damper input" (nviOAMinPos)	Optional	SNVT_lev_percent	input	Dynamic minimum position setpoint for an outdoor air damper. When valid it will supersede nciOAMinPos.

NV Member	Requirement	CNP Type	Direction	Description
60 "Minimum air flow setpoint input" (nviMinAirFlow)	Optional	SNVT_lev_percent	input	Dynamic minimum cooling air flow setpoint for single or dual duct VAV terminal units. When valid it will supersede nciMinFlow.
61 "Minimum heat air flow setpoint input" (nviMinAirFlowHt)	Optional	SNVT_lev_percent	input	Dynamic minimum heating air flow setpoint for single or dual duct VAV terminal units. When valid it will supersede nciMinFlowHeat.
62 "Air flow setpoint input" (nviAirFlowSetpt)	Optional	SNVT_flow	input	The controller will add this input with nciFlowOffset to derive the active flow setpoint
63 "Terminal load input" (nviTerminalLoad)	Optional	SNVT_lev_percent	input	When used with nvoTerminalLoad from another controller can be used to coordinate master/salve operation
64 "Mixed air temperature output" (nvoMixedAirTemp)	Optional	SNVT_temp_p	output	The temperature of the combined return and fresh airstreams in an AHU before they reach the water coils
65 "Local space temperature output" (nvoLocalSpaceTmp)	Optional	SNVT_temp_p	output	Local hardwired space temperature input
66 "Effective air flow heat setpoint output" (nvoEffFlowSPHeat)	Optional	SNVT_flow	output	The hot or ventilation duct flow setpoint of a dual duct unit.
67 "Flow control damper heat setpoint output" (nvoFlowSPHeat)	Optional	SNVT_lev_percent	output	Active flow setpoint used by the flow control loop for a hot or ventilation deck in a dual duct unit
68 "Air flow heat output" (nvoAirFlowHeat)	Optional	SNVT_flow	output	Air flow of a hot or ventilation deck of a dual duct VAV terminal
69 "HVAC saturation status" (nvoSatStatus)	Optional	SNVT_hvac_satsts	output	Indicates whether the control algorithm capacity limits, or end device physical limits, have been reached

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to SFPTsccRadiator	Mandatory	none	not permitted	Maximum period of time that expires before the specified NV outputs will automatically be updated
SCPTsetPnts applying to SFPTsccRadiator	Mandatory	none	not permitted	Space temperature setpoints for the various heat, cool, and occupancy modes
SCPTminSendTime applying to SFPTsccRadiator	Optional	none	not permitted	Minimum period of time between automatic NV output transmissions
SCPTmaxRcvTime applying to SFPTsccRadiator	Optional	none	not permitted	Maximum time that elapses after the last update to a specified NV input before the SCC starts to use its default values
SCPTlocation applying to SFPTsccRadiator	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTbypassTime applying to nviOccManCmd	Optional	none	not permitted	Maximum amount of time that the SCC can be in the bypass (occupancy) mode following a single bypass request
SCPTmanOvrTime applying to nviValveOverride	Optional	none	not permitted	Maximum time that the SCC will stay in a manual mode that was requested by a NV input, without receiving an update on that NV
SCPTminRnge applying to nvoOADamper	Optional	none	not permitted	Minimum position for the outdoor air damper
SCPTlimitCO2 applying to nviSpaceIAQ	Optional	none	not permitted	Maximum limit to allowable carbon dioxide within a defined area
SCPThumSetpt applying to nviSpaceRH	Optional	none	not permitted	High-limit humidity setpoint for the controlled space
SCPTnumValves applying to SFPTsccRadiator	Optional	none	not permitted	Selects whether the SCC is used in a two pipe (one valve) or four pipe (two valve) system
SCPTductArea applying to SFPTsccRadiator	Optional	none	not permitted	Nominal cross-sectional airflow area of a VAV terminal

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTnomAirFlow applying to nvoFlowSetpoint	Optional	none	not permitted	Nominal airflow volume of a VAV terminal
SCPTsensConstVAV applying to SFPTsccRadiator	Optional	none	not permitted	Calibrates the airflow reading of a VAV terminal
SCPTminFlow applying to nviAirflow	Optional	none	not permitted	Minimum airflow setpoint of a VAV terminal
SCPTmaxFlow applying to nviAirflow	Optional	none	not permitted	Maximum airflow setpoint of a VAV terminal
SCPTminFlowHeat applying to SFPTsccRadiator	Optional	none	not permitted	Minimum airflow setpoint of a VAV terminal while heating
SCPTmaxFlowHeat applying to SFPTsccRadiator	Optional	none	not permitted	Maximum airflow setpoint of a VAV terminal while heating
SCPTminFlowStby applying to SFPTsccRadiator	Optional	none	not permitted	Minimum airflow setpoint of a VAV terminal in the Standby (occupancy) mode
SCPThvacType applying to SFPTsccRadiator	Optional	device specific, constant	not permitted	This value is set by the manufacturer to allow an integrator to know the function of this SCC device
SCPTfanOperation applying to SFPTsccRadiator	Optional	none	not permitted	Specifies fan operation during occupied and occupied standby. Fan operation during unoccupied is manufacturer defined
SCPTminFlowUnit applying to SFPTsccRadiator	Optional	none	not permitted	Unit minimum air flow for dual duct VAV Terminal units
SCPTmaxFlowUnit applying to SFPTsccRadiator	Optional	none	not permitted	Unit maximum air flow for dual duct VAV Terminal units
SCPTminFlowHeatStby applying to SFPTsccRadiator	Optional	none	not permitted	The heating or ventilated deck minimum flow of a dual duct VAV Terminal unit during occupied standby mode

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTminFlowUnitStby applying to SFPTsccRadiator	Optional	none	not permitted	Total unit minimum airflow for dual duct units during occupied standby mode
SCPToffsetFlow applying to SFPTsccRadiator	Optional	none	not permitted	
SCPTareaDuctHeat applying to SFPTsccRadiator	Optional	none	not permitted	Nominal cross-sectional airflow area of the hot or ventilation deck of a dual duct VAV terminal unit
SCPTnomAirFlowHeat applying to SFPTsccRadiator	Optional	none	not permitted	Value used to provide the nominal airflow volume of a hot or ventilation deck of a dual duct VAV terminal
SCPTgainVAVHeat applying to SFPTsccRadiator	Optional	none	not permitted	Calibration constant used to calculate airflow
SCPTnumDampers applying to SFPTsccRadiator	Optional	none	not permitted	Indicates to the controller if it is in a single or dual duct system
SCPTminFlowUnitHeat applying to SFPTsccRadiator	Optional	none	not permitted	Minimum airflow setpoint of a single duct, or the unit minimum airflow setpoint of a dual duct VAV terminal when using a unit (local) heating source
SCPTsaturationDelay applying to SFPTsccRadiator	Optional	none	not permitted	

7.74 SFPTsccAHU (8508)

NV Member	Requirement	CNP Type	Direction	Description
1 "Space temperature input" (nviSpaceTemp)	Mandatory	SNVT_temp_p	input	Connects an external space temperature sensor to the node
2 "Temperature setpoint input (absolute)" (nviSetpoint)	Optional	SNVT_temp_p	input	Allows the temperature setpoints for the occupied and standby modes to be changed via the network
3 "Setpoint offset input" (nviSetptOffset)	Optional	SNVT_temp_p	input	Shifts the effective occupied and standby temperature setpoints by adding this NV's value to the present setpoints
4 "Setpoint shift input" (nviSetptShift)	Optional	SNVT_temp_setpt	input	Shifts the effective heat/cool setpoints by adding the corresponding value in this NV to the present setpoints
5 "Occupancy scheduler input" (nviOccSchedule)	Optional	SNVT_tod_event	input	Commands the SCC into different occupancy modes
6 "Occupancy override input" (nviOccManCmd)	Optional	SNVT_occupancy	input	Commands the SCC into different occupancy modes
7 "Occupancy sensor input" (nviOccSensor)	Optional	SNVT_occupancy	input	Indicates the presence of occupants in the controlled space
8 "Application mode input" (nviApplicMode)	Optional	SNVT_hvac_mode	input	Used to coordinate the SCC with any supervisory controller
9 "Heat/cool mode input" (nviHeatCool)	Optional	SNVT_hvac_mode	input	Used to coordinate the SCC with any node that may need to control the heat/cool changeover of the unit

NV Member	Requirement	CNP Type	Direction	Description
10 "Fan speed command input" (nviFanSpeedCmd)	Optional	SNVT_switch	input	Enables connection of an external fan speed switch, or allows a supervisory device to override the fan speed
11 "Compressor enable input" (nviComprEnable)	Optional	SNVT_switch	input	This input is used to disable compressor operation
12 "Auxiliary heat enable input" (nviAuxHeatEnable)	Optional	SNVT_switch	input	This input is used to disable auxiliary heat operation
13 "Economizer enable input" (nviEconEnable)	Optional	SNVT_switch	input	This input is used to enable and disable economizer operation
14 "Energy hold-off input" (nviEnergyHoldOff)	Optional	SNVT_switch	input	This input is used to stop heating and cooling while allowing the unit to protect the space from temperature extremes
15 "Water valve override input" (nviValveOverride)	Optional	SNVT_hvac_overid	input	Commands the controller into a manual mode for overriding water valves
16 "Airflow override input" (nviFlowOverride)	Optional	SNVT_hvac_overid	input	Commands the controller into a manual mode for overriding airflow control
17 "Emergency override input" (nviEmergOverride)	Optional	SNVT_hvac_emerg	input	Commands the device into different emergency modes
18 "Source temperature input" (nviSourceTemp)	Optional	SNVT_temp_p	input	Indicates the temperature of the air or water being supplied to the unit for heating and/or cooling capacity
19 "Outdoor air temperature input" (nviOutdoorTemp)	Optional	SNVT_temp_p	input	Represents information from an outdoor air temperature sensor

NV Member	Requirement	CNP Type	Direction	Description
20 "Space humidity input" (nviSpaceRH)	Optional	SNVT_lev_percent	input	Measured space humidity in percent
21 "Outdoor air humidity input" (nviOutdoorRH)	Optional	SNVT_lev_percent	input	Measured outdoor humidity in percent
22 "Space indoor air quality input" (nviSpaceIAQ)	Optional	SNVT_ppm	input	Measured space CO2 or VOC levels in PPM
23 "Space dewpoint temperature input" (nviSpaceDewPt)	Optional	SNVT_temp_p	input	Measured space dewpoint temperature
24 "Outdoor air dewpoint temperature input" (nviOutdoorDewPt)	Optional	SNVT_temp_p	input	Measured outdoor dewpoint temperature
25 "Airflow input" (nviAirflow)	Optional	SNVT_flow	input	The measured supply airflow value is typically provided by a flow sensor on the network
26 "Effective space temperature output" (nvoSpaceTemp)	Mandatory	SNVT_temp_p	output	Used to monitor the effective space temperature that the SCC is using for control
27 "Unit status output" (nvoUnitStatus)	Mandatory	SNVT_hvac_status	output	Reports the SCC status
28 "Effective setpoint output" (nvoEffectSetpt)	Optional	SNVT_temp_p	output	Monitors the effective temperature setpoint
29 "Effective occupancy output" (nvoEffectOccup)	Optional	SNVT_occupancy	output	Actual occupancy mode of the unit

NV Member	Requirement	CNP Type	Direction	Description
30 "Effective heat/cool output" (nvoHeatCool)	Optional	SNVT_hvac_mode	output	Actual heat/cool mode of the unit
31 "Local setpoint output" (nvoSetpoint)	Optional	SNVT_temp_p	output	Space temperature setpoint value if a setpoint device is hardwired
32 "Local setpoint shift output" (nvoSetptShift)	Optional	SNVT_temp_setpt	output	Locally determined shift of the effective heat/cool setpoints
33 "Fan speed output" (nvoFanSpeed)	Optional	SNVT_switch	output	Actual fan speed of a local multi-speed fan as well as the requested speed of a remote fan
34 "Discharge air temperature output" (nvoDischAirTemp)	Optional	SNVT_temp_p	output	Monitors the temperature of the air that leaves the SCC
35 "Absolute power consumption output" (nvoLoadAbs)	Optional	SNVT_power	output	Present power consumption of the unit
36 "Absolute power consumption (kW) output" (nvoLoadAbsK)	Optional	SNVT_power_kilo	output	Present power consumption of the unit
37 "Terminal load output" (nvoTerminalLoad)	Optional	SNVT_lev_percent	output	Present heat/cool energy demand of the unit
38 "Primary heat output" (nvoHeatPrimary)	Optional	SNVT_lev_percent	output	Present level of the primary heat output
39 "Secondary heat output" (nvoHeatSecondary)	Optional	SNVT_lev_percent	output	Present level of the secondary heat output

NV Member	Requirement	CNP Type	Direction	Description
40 "Primary cool output" (nvoCoolPrimary)	Optional	SNVT_lev_percent	output	Present level of the primary mechanical cooling output
41 "Secondary cool output" (nvoCoolSecondary)	Optional	SNVT_lev_percent	output	Present level of the secondary mechanical cooling output
42 "Outdoor air damper output" (nvoOADamper)	Optional	SNVT_lev_percent	output	Present position of the outdoor air damper (if hardwired) or as a request to a remote outdoor air damper
43 "Space humidity output" (nvoSpaceRH)	Optional	SNVT_lev_percent	output	Space humidity in percent, if the SCC Device has a hardwired humidity sensor
44 "Outdoor air humidity output" (nvoOutdoorRH)	Optional	SNVT_lev_percent	output	Outdoor air humidity in percent, if the SCC Device has a hardwired humidity sensor
45 "Outdoor air temperature output" (nvoOutdoorTemp)	Optional	SNVT_temp_p	output	Monitors the outdoor air temperature if the unit controller provides a hardwired temperature sensor
46 "Space CO2 sensor output" (nvoSpaceCO2)	Optional	SNVT_ppm	output	Space CO2 concentration in ppm, if the SCC Device has a hardwired CO2 sensor
47 "Space dewpoint temperature output" (nvoSpaceDewPt)	Optional	SNVT_temp_p	output	Space dewpoint temperature
48 "Humidifier output" (nvoHumidifier)	Optional	SNVT_lev_percent	output	Present value of the humidifier (if hardwired) or can be used to control a remote humidifier or control valve
49 "Energy hold-off output" (nvoEnergyHoldOff)	Optional	SNVT_switch	output	Present state of an energy hold-off device that is hardwired to the controller

NV Member	Requirement	CNP Type	Direction	Description
50 "Effective airflow setpoint output" (nvoEffectFlowSP)	Optional	SNVT_flow	output	Active flow setpoint used by the flow control loop
51 "Flow control damper setpoint output" (nvoFlowSetpoint)	Optional	SNVT_lev_percent	output	Active flow setpoint used by the flow control loop
52 "Airflow output" (nvoAirflow)	Optional	SNVT_flow	output	Measured airflow in the unit
53 "Heat source temperature input" (nviHeatSrcTemp)	Optional	SNVT_temp_p	input	Temperature of the air or water being supplied to the unit for heating capacity
54 "Cool source temperature input" (nviCoolSrcTemp)	Optional	SNVT_temp_p	input	Temperature of the air or water being supplied to the unit for cooling capacity
55 "Primary heat input for slave operation" (nviHeatPriSlave)	Optional	SNVT_lev_percent	input	This input NV is intended for slave operation
56 "Secondary heat input for slave operation" (nviHeatSecSlave)	Optional	SNVT_lev_percent	input	This input NV is intended for slave operation
57 "Primary cool input for slave operation" (nviCoolPriSlave)	Optional	SNVT_lev_percent	input	This input NV is intended for slave operation
58 "Secondary cool input for slave operation" (nviCoolSecSlave)	Optional	SNVT_lev_percent	input	This input NV is intended for slave operation
59 "Minimum position OA damper input" (nviOAMinPos)	Optional	SNVT_lev_percent	input	Dynamic minimum position setpoint for an outdoor air damper. When valid it will supersede nciOAMinPos.

NV Member	Requirement	CNP Type	Direction	Description
60 "Minimum air flow setpoint input" (nviMinAirFlow)	Optional	SNVT_lev_percent	input	Dynamic minimum cooling airflow setpoint for single or dual duct VAV terminal units. When valid it will supersede nciMinFlow.
61 "Minimum heat air flow setpoint input" (nviMinAirFlowHt)	Optional	SNVT_lev_percent	input	Dynamic minimum heating airflow setpoint for single or dual duct VAV terminal units. When valid it will supersede nciMinFlowHeat.
62 "Air flow setpoint input" (nviAirFlowSetpt)	Optional	SNVT_flow	input	The controller will add this input with nciFlowOffset to derive the active flow setpoint
63 "Terminal load input" (nviTerminalLoad)	Optional	SNVT_lev_percent	input	When used with nvoTerminalLoad from another controller can be used to coordinate master/salve operation
64 "Mixed air temperature output" (nvoMixedAirTemp)	Optional	SNVT_temp_p	output	The temperature of the combined return and fresh airstreams in an AHU before they reach the water coils
65 "Local space temperature output" (nvoLocalSpaceTmp)	Optional	SNVT_temp_p	output	Local hardwired space temperature input
66 "Effective air flow heat setpoint output" (nvoEffFlowSPHeat)	Optional	SNVT_flow	output	The hot or ventilation duct flow setpoint of a dual duct unit.
67 "Flow control damper heat setpoint output" (nvoFlowSPHeat)	Optional	SNVT_lev_percent	output	Active flow setpoint used by the flow control loop for a hot or ventilation deck in a dual duct unit
68 "Air flow heat output" (nvoAirFlowHeat)	Optional	SNVT_flow	output	Air flow of a hot or ventilation deck of a dual duct VAV terminal
69 "HVAC saturation status" (nvoSatStatus)	Optional	SNVT_hvac_satsts	output	Indicates whether the control algorithm capacity limits, or end device physical limits, have been reached

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to SFPTsccAHU	Mandatory	none	not permitted	Maximum period of time that expires before the specified NV outputs will automatically be updated
SCPTsetPnts applying to SFPTsccAHU	Mandatory	none	not permitted	Space temperature setpoints for the various heat, cool, and occupancy modes
SCPTminSendTime applying to SFPTsccAHU	Optional	none	not permitted	Minimum period of time between automatic NV output transmissions
SCPTmaxRcvTime applying to SFPTsccAHU	Optional	none	not permitted	Maximum time that elapses after the last update to a specified NV input before the SCC starts to use its default values
SCPTlocation applying to SFPTsccAHU	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTbypassTime applying to nviOccManCmd	Optional	none	not permitted	Maximum amount of time that the SCC can be in the bypass (occupancy) mode following a single bypass request
SCPTmanOvrTime applying to nviValveOverride	Optional	none	not permitted	Maximum time that the SCC will stay in a manual mode that was requested by a NV input, without receiving an update on that NV
SCPTminRnge applying to nvoOADamper	Optional	none	not permitted	Minimum position for the outdoor air damper
SCPTlimitCO2 applying to nviSpaceIAQ	Optional	none	not permitted	Maximum limit to allowable carbon dioxide within a defined area
SCPThumSetpt applying to nviSpaceRH	Optional	none	not permitted	High-limit humidity setpoint for the controlled space
SCPTnumValves applying to SFPTsccAHU	Optional	none	not permitted	Selects whether the SCC is used in a two pipe (one valve) or four pipe (two valve) system
SCPTductArea applying to SFPTsccAHU	Optional	none	not permitted	Nominal cross-sectional airflow area of a VAV terminal

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTnomAirFlow applying to nvoFlowSetpoint	Optional	none	not permitted	Nominal airflow volume of a VAV terminal
SCPTsensConstVAV applying to SFPTsccAHU	Optional	none	not permitted	Calibrates the airflow reading of a VAV terminal
SCPTminFlow applying to nviAirflow	Optional	none	not permitted	Minimum airflow setpoint of a VAV terminal
SCPTmaxFlow applying to nviAirflow	Optional	none	not permitted	Maximum airflow setpoint of a VAV terminal
SCPTminFlowHeat applying to SFPTsccAHU	Optional	none	not permitted	Minimum airflow setpoint of a VAV terminal while heating
SCPTmaxFlowHeat applying to SFPTsccAHU	Optional	none	not permitted	Maximum airflow setpoint of a VAV terminal while heating
SCPTminFlowStby applying to SFPTsccAHU	Optional	none	not permitted	Minimum airflow setpoint of a VAV terminal in the Standby (occupancy) mode
SCPThvacType applying to SFPTsccAHU	Optional	device specific, constant	not permitted	This value is set by the manufacturer to allow an integrator to know the function of this SCC device
SCPTfanOperation applying to SFPTsccAHU	Optional	none	not permitted	Specifies fan operation during occupied and occupied standby. Fan operation during unoccupied is manufacturer defined
SCPTminFlowUnit applying to SFPTsccAHU	Optional	none	not permitted	Unit minimum air flow for dual duct VAV Terminal units
SCPTmaxFlowUnit applying to SFPTsccAHU	Optional	none	not permitted	Unit maximum air flow for dual duct VAV Terminal units
SCPTminFlowHeatStby applying to SFPTsccAHU	Optional	none	not permitted	The heating or ventilated deck minimum flow of a dual duct VAV Terminal unit during occupied standby mode

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTminFlowUnitStby applying to SFPTsccAHU	Optional	none	not permitted	Total unit minimum airflow for dual duct units during occupied standby mode
SCPToffsetFlow applying to SFPTsccAHU	Optional	none	not permitted	
SCPTareaDuctHeat applying to SFPTsccAHU	Optional	none	not permitted	Nominal cross-sectional airflow area of the hot or ventilation deck of a dual duct VAV terminal unit
SCPTnomAirFlowHeat applying to SFPTsccAHU	Optional	none	not permitted	Value used to provide the nominal airflow volume of a hot or ventilation deck of a dual duct VAV terminal
SCPTgainVAVHeat applying to SFPTsccAHU	Optional	none	not permitted	Calibration constant used to calculate airflow
SCPTnumDampers applying to SFPTsccAHU	Optional	none	not permitted	Indicates to the controller if it is in a single or dual duct system
SCPTminFlowUnitHeat applying to SFPTsccAHU	Optional	none	not permitted	Minimum airflow setpoint of a single duct, or the unit minimum airflow setpoint of a dual duct VAV terminal when using a unit (local) heating source
SCPTsaturationDelay applying to SFPTsccAHU	Optional	none	not permitted	

7.75 SFPTsccSelfContained (8509)

NV Member	Requirement	CNP Type	Direction	Description
1 "Space temperature input" (nviSpaceTemp)	Mandatory	SNVT_temp_p	input	Connects an external space temperature sensor to the node
2 "Temperature setpoint input (absolute)" (nviSetpoint)	Optional	SNVT_temp_p	input	Allows the temperature setpoints for the occupied and standby modes to be changed via the network
3 "Setpoint offset input" (nviSetptOffset)	Optional	SNVT_temp_p	input	Shifts the effective occupied and standby temperature setpoints by adding this NV's value to the present setpoints
4 "Setpoint shift input" (nviSetptShift)	Optional	SNVT_temp_setpt	input	Shifts the effective heat/cool setpoints by adding the corresponding value in this NV to the present setpoints
5 "Occupancy scheduler input" (nviOccSchedule)	Optional	SNVT_tod_event	input	Commands the SCC into different occupancy modes
6 "Occupancy override input" (nviOccManCmd)	Optional	SNVT_occupancy	input	Commands the SCC into different occupancy modes
7 "Occupancy sensor input" (nviOccSensor)	Optional	SNVT_occupancy	input	Indicates the presence of occupants in the controlled space
8 "Application mode input" (nviApplicMode)	Optional	SNVT_hvac_mode	input	Used to coordinate the SCC with any supervisory controller
9 "Heat/cool mode input" (nviHeatCool)	Optional	SNVT_hvac_mode	input	Used to coordinate the SCC with any node that may need to control the heat/cool changeover of the unit

NV Member	Requirement	CNP Type	Direction	Description
10 "Fan speed command input" (nviFanSpeedCmd)	Optional	SNVT_switch	input	Enables connection of an external fan speed switch, or allows a supervisory device to override the fan speed
11 "Compressor enable input" (nviComprEnable)	Optional	SNVT_switch	input	This input is used to disable compressor operation
12 "Auxiliary heat enable input" (nviAuxHeatEnable)	Optional	SNVT_switch	input	This input is used to disable auxiliary heat operation
13 "Economizer enable input" (nviEconEnable)	Optional	SNVT_switch	input	This input is used to enable and disable economizer operation
14 "Energy hold-off input" (nviEnergyHoldOff)	Optional	SNVT_switch	input	This input is used to stop heating and cooling while allowing the unit to protect the space from temperature extremes
15 "Water valve override input" (nviValveOverride)	Optional	SNVT_hvac_overid	input	Commands the controller into a manual mode for overriding water valves
16 "Airflow override input" (nviFlowOverride)	Optional	SNVT_hvac_overid	input	Commands the controller into a manual mode for overriding airflow control
17 "Emergency override input" (nviEmergOverride)	Optional	SNVT_hvac_emerg	input	Commands the device into different emergency modes
18 "Source temperature input" (nviSourceTemp)	Optional	SNVT_temp_p	input	Indicates the temperature of the air or water being supplied to the unit for heating and/or cooling capacity
19 "Outdoor air temperature input" (nviOutdoorTemp)	Optional	SNVT_temp_p	input	Represents information from an outdoor air temperature sensor

NV Member	Requirement	CNP Type	Direction	Description
20 "Space humidity input" (nviSpaceRH)	Optional	SNVT_lev_percent	input	Measured space humidity in percent
21 "Outdoor air humidity input" (nviOutdoorRH)	Optional	SNVT_lev_percent	input	Measured outdoor humidity in percent
22 "Space indoor air quality input" (nviSpaceIAQ)	Optional	SNVT_ppm	input	Measured space CO2 or VOC levels in PPM
23 "Space dewpoint temperature input" (nviSpaceDewPt)	Optional	SNVT_temp_p	input	Measured space dewpoint temperature
24 "Outdoor air dewpoint temperature input" (nviOutdoorDewPt)	Optional	SNVT_temp_p	input	Measured outdoor dewpoint temperature
25 "Airflow input" (nviAirflow)	Optional	SNVT_flow	input	The measured supply airflow value is typically provided by a flow sensor on the network
26 "Effective space temperature output" (nvoSpaceTemp)	Mandatory	SNVT_temp_p	output	Used to monitor the effective space temperature that the SCC is using for control
27 "Unit status output" (nvoUnitStatus)	Mandatory	SNVT_hvac_status	output	Reports the SCC status
28 "Effective setpoint output" (nvoEffectSetpt)	Optional	SNVT_temp_p	output	Monitors the effective temperature setpoint
29 "Effective occupancy output" (nvoEffectOccup)	Optional	SNVT_occupancy	output	Actual occupancy mode of the unit

NV Member	Requirement	CNP Type	Direction	Description
30 "Effective heat/cool output" (nvoHeatCool)	Optional	SNVT_hvac_mode	output	Actual heat/cool mode of the unit
31 "Local setpoint output" (nvoSetpoint)	Optional	SNVT_temp_p	output	Space temperature setpoint value if a setpoint device is hardwired
32 "Local setpoint shift output" (nvoSetptShift)	Optional	SNVT_temp_setpt	output	Locally determined shift of the effective heat/cool setpoints
33 "Fan speed output" (nvoFanSpeed)	Optional	SNVT_switch	output	Actual fan speed of a local multi-speed fan as well as the requested speed of a remote fan
34 "Discharge air temperature output" (nvoDischAirTemp)	Optional	SNVT_temp_p	output	Monitors the temperature of the air that leaves the SCC
35 "Absolute power consumption output" (nvoLoadAbs)	Optional	SNVT_power	output	Present power consumption of the unit
36 "Absolute power consumption (kW) output" (nvoLoadAbsK)	Optional	SNVT_power_kilo	output	Present power consumption of the unit
37 "Terminal load output" (nvoTerminalLoad)	Optional	SNVT_lev_percent	output	Present heat/cool energy demand of the unit
38 "Primary heat output" (nvoHeatPrimary)	Optional	SNVT_lev_percent	output	Present level of the primary heat output
39 "Secondary heat output" (nvoHeatSecondary)	Optional	SNVT_lev_percent	output	Present level of the secondary heat output

NV Member	Requirement	CNP Type	Direction	Description
40 "Primary cool output" (nvoCoolPrimary)	Optional	SNVT_lev_percent	output	Present level of the primary mechanical cooling output
41 "Secondary cool output" (nvoCoolSecondary)	Optional	SNVT_lev_percent	output	Present level of the secondary mechanical cooling output
42 "Outdoor air damper output" (nvoOADamper)	Optional	SNVT_lev_percent	output	Present position of the outdoor air damper (if hardwired) or as a request to a remote outdoor air damper
43 "Space humidity output" (nvoSpaceRH)	Optional	SNVT_lev_percent	output	Space humidity in percent, if the SCC Device has a hardwired humidity sensor
44 "Outdoor air humidity output" (nvoOutdoorRH)	Optional	SNVT_lev_percent	output	Outdoor air humidity in percent, if the SCC Device has a hardwired humidity sensor
45 "Outdoor air temperature output" (nvoOutdoorTemp)	Optional	SNVT_temp_p	output	Monitors the outdoor air temperature if the unit controller provides a hardwired temperature sensor
46 "Space CO2 sensor output" (nvoSpaceCO2)	Optional	SNVT_ppm	output	Space CO2 concentration in ppm, if the SCC Device has a hardwired CO2 sensor
47 "Space dewpoint temperature output" (nvoSpaceDewPt)	Optional	SNVT_temp_p	output	Space dewpoint temperature
48 "Humidifier output" (nvoHumidifier)	Optional	SNVT_lev_percent	output	Present value of the humidifier (if hardwired) or can be used to control a remote humidifier or control valve
49 "Energy hold-off output" (nvoEnergyHoldOff)	Optional	SNVT_switch	output	Present state of an energy hold-off device that is hardwired to the controller

NV Member	Requirement	CNP Type	Direction	Description
50 "Effective airflow setpoint output" (nvoEffectFlowSP)	Optional	SNVT_flow	output	Active flow setpoint used by the flow control loop
51 "Flow control damper setpoint output" (nvoFlowSetpoint)	Optional	SNVT_lev_percent	output	Active flow setpoint used by the flow control loop
52 "Airflow output" (nvoAirflow)	Optional	SNVT_flow	output	Measured airflow in the unit
53 "Heat source temperature input" (nviHeatSrcTemp)	Optional	SNVT_temp_p	input	Temperature of the air or water being supplied to the unit for heating capacity
54 "Cool source temperature input" (nviCoolSrcTemp)	Optional	SNVT_temp_p	input	Temperature of the air or water being supplied to the unit for cooling capacity
55 "Primary heat input for slave operation" (nviHeatPriSlave)	Optional	SNVT_lev_percent	input	This input NV is intended for slave operation
56 "Secondary heat input for slave operation" (nviHeatSecSlave)	Optional	SNVT_lev_percent	input	This input NV is intended for slave operation
57 "Primary cool input for slave operation" (nviCoolPriSlave)	Optional	SNVT_lev_percent	input	This input NV is intended for slave operation
58 "Secondary cool input for slave operation" (nviCoolSecSlave)	Optional	SNVT_lev_percent	input	This input NV is intended for slave operation
59 "Minimum position OA damper input" (nviOAMinPos)	Optional	SNVT_lev_percent	input	Dynamic minimum position setpoint for an outdoor air damper. When valid it will supersede nciOAMinPos.

NV Member	Requirement	CNP Type	Direction	Description
60 "Minimum air flow setpoint input" (nviMinAirFlow)	Optional	SNVT_lev_percent	input	Dynamic minimum cooling airflow setpoint for single or dual duct VAV terminal units. When valid it will supersede nciMinFlow.
61 "Minimum heat air flow setpoint input" (nviMinAirFlowHt)	Optional	SNVT_lev_percent	input	Dynamic minimum heating airflow setpoint for single or dual duct VAV terminal units. When valid it will supersede nciMinFlowHeat.
62 "Air flow setpoint input" (nviAirFlowSetpt)	Optional	SNVT_flow	input	The controller will add this input with nciFlowOffset to derive the active flow setpoint
63 "Terminal load input" (nviTerminalLoad)	Optional	SNVT_lev_percent	input	When used with nvoTerminalLoad from another controller can be used to coordinate master/salve operation
64 "Mixed air temperature output" (nvoMixedAirTemp)	Optional	SNVT_temp_p	output	The temperature of the combined return and fresh airstreams in an AHU before they reach the water coils
65 "Local space temperature output" (nvoLocalSpaceTmp)	Optional	SNVT_temp_p	output	Local hardwired space temperature input
66 "Effective air flow heat setpoint output" (nvoEffFlowSPHeat)	Optional	SNVT_flow	output	The hot or ventilation duct flow setpoint of a dual duct unit.
67 "Flow control damper heat setpoint output" (nvoFlowSPHeat)	Optional	SNVT_lev_percent	output	Active flow setpoint used by the flow control loop for a hot or ventilation deck in a dual duct unit
68 "Air flow heat output" (nvoAirFlowHeat)	Optional	SNVT_flow	output	Air flow of a hot or ventilation deck of a dual duct VAV terminal
69 "HVAC saturation status" (nvoSatStatus)	Optional	SNVT_hvac_satsts	output	Indicates whether the control algorithm capacity limits, or end device physical limits, have been reached

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to SFPTsccSelfContained	Mandatory	none	not permitted	Maximum period of time that expires before the specified NV outputs will automatically be updated
SCPTsetPnts applying to SFPTsccSelfContained	Mandatory	none	not permitted	Space temperature setpoints for the various heat, cool, and occupancy modes
SCPTminSendTime applying to SFPTsccSelfContained	Optional	none	not permitted	Minimum period of time between automatic NV output transmissions
SCPTmaxRcvTime applying to SFPTsccSelfContained	Optional	none	not permitted	Maximum time that elapses after the last update to a specified NV input before the SCC starts to use its default values
SCPTlocation applying to SFPTsccSelfContained	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTbypassTime applying to nviOccManCmd	Optional	none	not permitted	Maximum amount of time that the SCC can be in the bypass (occupancy) mode following a single bypass request
SCPTmanOvrTime applying to nviValveOverride	Optional	none	not permitted	Maximum time that the SCC will stay in a manual mode that was requested by a NV input, without receiving an update on that NV
SCPTminRnge applying to nvoOADamper	Optional	none	not permitted	Minimum position for the outdoor air damper
SCPTlimitCO2 applying to nviSpaceIAQ	Optional	none	not permitted	Maximum limit to allowable carbon dioxide within a defined area
SCPThumSetpt applying to nviSpaceRH	Optional	none	not permitted	High-limit humidity setpoint for the controlled space

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTnumValves applying to SFPTsccSelfContained	Optional	none	not permitted	Selects whether the SCC is used in a two pipe (one valve) or four pipe (two valve) system
SCPTductArea applying to SFPTsccSelfContained	Optional	none	not permitted	Nominal cross-sectional airflow area of a VAV terminal
SCPTnomAirFlow applying to nvoFlowSetpoint	Optional	none	not permitted	Nominal airflow volume of a VAV terminal
SCPTsensConstVAV applying to SFPTsccSelfContained	Optional	none	not permitted	Calibrates the airflow reading of a VAV terminal
SCPTminFlow applying to nviAirflow	Optional	none	not permitted	Minimum airflow setpoint of a VAV terminal
SCPTmaxFlow applying to nviAirflow	Optional	none	not permitted	Maximum airflow setpoint of a VAV terminal
SCPTminFlowHeat applying to SFPTsccSelfContained	Optional	none	not permitted	Minimum airflow setpoint of a VAV terminal while heating
SCPTmaxFlowHeat applying to SFPTsccSelfContained	Optional	none	not permitted	Maximum airflow setpoint of a VAV terminal while heating
SCPTminFlowStby applying to SFPTsccSelfContained	Optional	none	not permitted	Minimum airflow setpoint of a VAV terminal in the Standby (occupancy) mode
SCPThvacType applying to SFPTsccSelfContained	Optional	device specific, constant	not permitted	This value is set by the manufacturer to allow an integrator to know the function of this SCC device

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTfanOperation applying to SFPTsccSelfContained	Optional	none	not permitted	Specifies fan operation during occupied and occupied standby. Fan operation during unoccupied is manufacturer defined
SCPTminFlowUnit applying to SFPTsccSelfContained	Optional	none	not permitted	Unit minimum air flow for dual duct VAV Terminal units
SCPTmaxFlowUnit applying to SFPTsccSelfContained	Optional	none	not permitted	Unit maximum air flow for dual duct VAV Terminal units
SCPTminFlowHeatStby applying to SFPTsccSelfContained	Optional	none	not permitted	The heating or ventilated deck minimum flow of a dual duct VAV Terminal unit during occupied standby mode
SCPTminFlowUnitStby applying to SFPTsccSelfContained	Optional	none	not permitted	Total unit minimum airflow for dual duct units during occupied standby mode
SCPToffsetFlow applying to SFPTsccSelfContained	Optional	none	not permitted	
SCPTareaDuctHeat applying to SFPTsccSelfContained	Optional	none	not permitted	Nominal cross-sectional airflow area of the hot or ventilation deck of a dual duct VAV terminal unit
SCPTnomAirFlowHeat applying to SFPTsccSelfContained	Optional	none	not permitted	Value used to provide the nominal airflow volume of a hot or ventilation deck of a dual duct VAV terminal
SCPTgainVAVHeat applying to SFPTsccSelfContained	Optional	none	not permitted	Calibration constant used to calculate airflow
SCPTnumDampers applying to SFPTsccSelfContained	Optional	none	not permitted	Indicates to the controller if it is in a single or dual duct system

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTminFlowUnitHeat applying to SFPTsccSelfContained	Optional	none	not permitted	Minimum airflow setpoint of a single duct, or the unit minimum airflow setpoint of a dual duct VAV terminal when using a unit (local) heating source
SCPTsaturationDelay applying to SFPTsccSelfContained	Optional	none	not permitted	

7.76 SFPTdischargeAirController (8610)

NV Member	Requirement	CNP Type	Direction	Description
1 "Occupancy scheduler input" (nviOccSchedule)	Mandatory	SNVT_tod_event	input	Commands the Discharge-Air Controller into different occupancy modes typically sent by scheduler or supervisory node
2 "Occupancy override input" (nviOccManCmd)	Optional	SNVT_occupancy	input	Commands the Discharge-Air Controller into different occupancy modes
3 "Application mode input" (nviApplicMode)	Optional	SNVT_hvac_mode	input	Coordinates the Discharge-Air Controller with any supervisory controller
4 "Emergency override input" (nviEmergOverride)	Optional	SNVT_hvac_emerg	input	Commands the device into different emergency modes
5 "Duct static pressure input" (nviDuctStatPress)	Optional	SNVT_press_p	input	Connects a duct static pressure sensor or network output from another controller
6 "Duct static pressure setpoint input" (nviDuctStaticSP)	Optional	SNVT_press_p	input	Sets duct static pressure setpoint of the controller via the network

NV Member	Requirement	CNP Type	Direction	Description
7 "Discharge air cooling setpoint input" (nviDAClSP)	Optional	SNVT_temp_p	input	Sets discharge-air cooling setpoint of the controller via the network
8 "Discharge-Air heating setpoint input" (nviDAHtSP)	Optional	SNVT_temp_p	input	Sets discharge-air heating setpoint of the controller via the network
9 "Supply fan capacity input" (nviSupFanCap)	Optional	SNVT_lev_percent	input	Commands override of the supply fan capacity from another controller
10 "Exhaust fan capacity input" (nviExhFanCap)	Optional	SNVT_lev_percent	input	Connects network output from another controller to override the local exhaust fan capacity control
11 "Return fan capacity input" (nviRetFanCap)	Optional	SNVT_lev_percent	input	Commands an override of return fan capacity from another controller
12 "Fan differential setpoint input" (nviFanDiffSP)	Optional	SNVT_lev_percent	input	Setpoint for the percent capacity difference between the supply and return fans
13 "Building static pressure input" (nviBldgStatPress)	Optional	SNVT_press_p	input	Connects network building static pressure sensor or network output from another controller
14 "Building static pressure setpoint input" (nviBldgStaticSP)	Optional	SNVT_press_p	input	Connects network output from another controller to provide the building static pressure setpoint
15 "Primary cool enable input" (nviPriCoolEnable)	Optional	SNVT_switch	input	Connect network output from another controller to enable/disable the primary cooling output of the unit
16 "Primary heat enable input" (nviPriHeatEnable)	Optional	SNVT_switch	input	Connects network output from another controller to enable/disable the primary heating outputs of the unit

NV Member	Requirement	CNP Type	Direction	Description
17 "Economizer enable input" (nviEconEnable)	Optional	SNVT_switch	input	Enables and disables economizer operation
18 "Outdoor air minimum position input" (nviOAMinPos)	Optional	SNVT_lev_percent	input	Sets minimum outdoor air damper position of controller via the network
19 "Minimum outdoor airflow setpoint input" (nviMinOAFlowSP)	Optional	SNVT_flow	input	Commands a minimum outdoor airflow rate setpoint from the network
20 "Outdoor air temperature input" (nviOutdoorTemp)	Optional	SNVT_temp_p	input	Measured outdoor air dry bulb temperature
21 "Outdoor air humidity input" (nviOutdoorRH)	Optional	SNVT_lev_percent	input	Measured outdoor air humidity in percent
22 "Outdoor air enthalpy input" (nviOAEnthalpy)	Optional	SNVT_enthalpy	input	Connects a network outdoor air enthalpy sensor or network controller output
23 "Mixed air temperature setpoint input" (nviMATSP)	Optional	SNVT_temp_p	input	Commands a mixed air temperature setpoint from the network
24 "Return air temperature input" (nviRATemp)	Optional	SNVT_temp_p	input	Connects a network return air temperature sensor or network output from another controller
25 "Space enthalpy input" (nviSpaceEnthalpy)	Optional	SNVT_enthalpy	input	Connects a network return air or space enthalpy sensor or network output from another controller
26 "Space temperature input" (nviSpaceTemp)	Optional	SNVT_temp_p	input	Connects a network space temperature sensor or network output from another controller

NV Member	Requirement	CNP Type	Direction	Description
27 "Space humidity input" (nviSpaceRH)	Optional	SNVT_lev_percent	input	Connects a network return air or space relative humidity sensor or network output from another controller
28 "Humidification enable input" (nviHumEnable)	Optional	SNVT_switch	input	Enables humidification function in the controller
29 "Space humidification setpoint input" (nviSpaceHumSP)	Optional	SNVT_lev_percent	input	Connects a network space humidity setpoint or network output from another controller
30 "Dehumidification enable input" (nviDehumEnable)	Optional	SNVT_switch	input	Enables dehumidification function in the controller
31 "Space dehumidification setpoint input" (nviSpaceDehumSP)	Optional	SNVT_lev_percent	input	Connects a network space dehumidification setpoint or network output from another controller
32 "Discharge air dewpoint setpoint input" (nviDADewPointSP)	Optional	SNVT_temp_p	input	Commands a discharge-air dewpoint setpoint from the network
33 "Condenser water temperature input" (nviCWTemp)	Optional	SNVT_temp_p	input	Connects a network condenser water temperature sensor or network output from another controller
34 "Condenser water flow input" (nviCWFlow)	Optional	SNVT_switch	input	System condenser flow status
35 "Discharge air temperature output" (nvoDischAirTemp)	Mandatory	SNVT_temp_p	output	Monitors discharge-air temperature measured by a hardwired sensor
36 "Unit status output" (nvoUnitStatus)	Mandatory	SNVT_hvac_status	output	Reports the Discharge-Air Controller status

NV Member	Requirement	CNP Type	Direction	Description
37 "Effective discharge-air temperature setpoint output" (nvoEffDATempSP)	Optional	SNVT_temp_p	output	Monitors the effective discharge-air temperature setpoint the Discharge-Air Controller is using for control
38 "Duct static pressure output" (nvoDuctStatPress)	Optional	SNVT_press_p	output	Monitors the effective duct static pressure the controller is using for control
39 "Effective duct static pressure setpoint output" (nvoEffDuctStatSP)	Optional	SNVT_press_p	output	Monitors the effective duct static pressure setpoint the Discharge-Air Controller is using for control
40 "Effective heat/cool output" (nvoHeatCool)	Optional	SNVT_hvac_mode	output	Actual heat/cool mode of the unit
41 "Application mode output" (nvoApplicMode)	Optional	SNVT_hvac_mode	output	Used to control the mode of other controllers such as a VAV box controller
42 "Effective occupancy output" (nvoEffectOccup)	Optional	SNVT_occupancy	output	Transmits the current occupancy mode of the Discharge-Air Controller for monitoring
43 "Supply fan status output" (nvoSupFanStatus)	Optional	SNVT_switch	output	Actual status of the supply fan for monitoring
44 "Supply fan on/off control output" (nvoSupFanOnOff)	Optional	SNVT_switch	output	Used to start and stop the supply fan
45 "Supply fan capacity output" (nvoSupFanCap)	Optional	SNVT_lev_percent	output	Used to command the supply fan speed or capacity
46 "Exhaust fan status output" (nvoExhFanStatus)	Optional	SNVT_switch	output	Actual status of the exhaust fan for monitoring

NV Member	Requirement	CNP Type	Direction	Description
47 "Exhaust fan on/off control output" (nvoExhFanOnOff)	Optional	SNVT_switch	output	Used to start and stop the exhaust fan
48 "Exhaust fan capacity output" (nvoExhFanCap)	Optional	SNVT_lev_percent	output	Used to command the exhaust fan speed or capacity
49 "Exhaust damper control output" (nvoExhDamper)	Optional	SNVT_lev_percent	output	Present status of Exhaust Damper output for monitoring or control
50 "Return fan status output" (nvoRetFanStatus)	Optional	SNVT_switch	output	Actual status of the return fan for monitoring
51 "Return fan on/off control output" (nvoRetFanOnOff)	Optional	SNVT_switch	output	Used to start and stop the return fan
52 "Return fan capacity output" (nvoRetFanCap)	Optional	SNVT_lev_percent	output	Used to command the return fan speed or capacity
53 "Return fan pressure output" (nvoRetFanPress)	Optional	SNVT_press_p	output	Present value of return fan static pressure for monitoring
54 "Building static pressure output" (nvoBldgStatPress)	Optional	SNVT_press_p	output	Present value of the building static pressure for monitoring
55 "Economizer enabled output" (nvoEconEnabled)	Optional	SNVT_switch	output	Present enable/disable status of economizer for monitoring
56 "Outdoor air damper output" (nvoOADamper)	Optional	SNVT_lev_percent	output	Present level of the outdoor air damper or injection fan capacity output for monitoring or control

NV Member	Requirement	CNP Type	Direction	Description
57 "Outdoor airflow output" (nvoOAFlow)	Optional	SNVT_flow	output	Present value of the outdoor airflow for monitoring
58 "Local outdoor air temperature output" (nvoLocalOATemp)	Optional	SNVT_temp_p	output	Indicates value of a hardwired outdoor air temperature sensor
59 "Outdoor air temperature output" (nvoOutdoorTemp)	Optional	SNVT_temp_p	output	Present value of outdoor air temperature for monitoring
60 "Local outdoor air humidity output" (nvoLocalOARH)	Optional	SNVT_lev_percent	output	Indicates value of hardwired outdoor air relative humidity sensor
61 "Outdoor air humidity output" (nvoOutdoorRH)	Optional	SNVT_lev_percent	output	Present value of outdoor air humidity for monitoring
62 "Outdoor air enthalpy output" (nvoOAEnthalpy)	Optional	SNVT_enthalpy	output	Present value of the outdoor air enthalpy
63 "Primary Cooling Output" (nvoCoolPrimary)	Optional	SNVT_lev_percent	output	Present level of the primary cooling capacity
64 "Primary heating output" (nvoHeatPrimary)	Optional	SNVT_lev_percent	output	Present value of the primary heating capacity
65 "Mixed air temperature output" (nvoMATemp)	Optional	SNVT_temp_p	output	Present value of the mixed air dry bulb temperature
66 "Space temperature output" (nvoSpaceTemp)	Optional	SNVT_temp_p	output	Present value of the space temperature for monitoring

NV Member	Requirement	CNP Type	Direction	Description
67 "Return air temperature output" (nvoRATemp)	Optional	SNVT_temp_p	output	Present value of return air temperature for monitoring
68 "Space humidity output" (nvoSpaceRH)	Optional	SNVT_lev_percent	output	Present value of the space relative humidity for monitoring
69 "Space enthalpy output" (nvoSpaceEnthalpy)	Optional	SNVT_enthalpy	output	Present value of the space enthalpy
70 "Effective space humidification setpoint output" (nvoEffSpaceHumSP)	Optional	SNVT_lev_percent	output	Effective space low limit humidity setpoint for monitoring
71 "Humidification status output" (nvoHumidifier)	Optional	SNVT_lev_percent	output	Present level of the humidifier output for monitoring
72 "Effective space dehumidification setpoint output" (nvoEffSpaceDHSP)	Optional	SNVT_lev_percent	output	Effective space high limit humidity setpoint for monitoring
73 "Dehumidification status output" (nvoDehumidifier)	Optional	SNVT_switch	output	Present status of dehumidification control for monitoring
74 "Effective discharge-air dewpoint setpoint output" (nvoEffDADewPtSP)	Optional	SNVT_temp_p	output	Monitors the effective discharge-air dewpoint setpoint that the discharge-air controller is using for control
75 "Discharge air dewpoint temperature output" (nvoDADewPoint)	Optional	SNVT_temp_p	output	Present value of the discharge-air dewpoint temperature

NV Member	Requirement	CNP Type	Direction	Description
76 "Condenser capacity output" (nvoCondCap)	Optional	SNVT_lev_percent	output	Present value of the condenser capacity control output for monitoring
77 "Local condenser water temperature output" (nvoLocalCWTemp)	Optional	SNVT_temp_p	output	Transmits value of hardwired condenser water temperature sensor
78 "Condenser water temperature output" (nvoCWTemp)	Optional	SNVT_temp_p	output	Present value of condenser water temperature for monitoring
79 "Condenser water flow output" (nvoCWFlow)	Optional	SNVT_switch	output	Transmits current status of condenser water flow sensor for monitoring
80 "Condenser water pump output" (nvoCWPump)	Optional	SNVT_switch	output	Transmits the current state of condenser water pump output for monitoring or control

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to SFPTdischargeAirController	Mandatory	none	not permitted	Maximum period of time that expires before specified NV outputs will be automatically updated
SCPTdischargeAirCoolingSetpoint applying to SFPTdischargeAirController	Mandatory	none	not permitted	Default discharge-air cooling setpoint for the Discharge-Air Controller
SCPTdischargeAirHeatingSetpoint applying to SFPTdischargeAirController	Mandatory	none	not permitted	Default discharge-air heating setpoint for the Discharge-Air Controller
SCPTsetPnts applying to SFPTdischargeAirController	Optional	none	not permitted	Space temperature setpoints for various heat, cool and occupancy modes

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTminSendTime applying to SFPTdischargeAirController	Optional	none	not permitted	Minimum time between automatic NV output transmissions
SCPTmaxRcvTime applying to SFPTdischargeAirController	Optional	none	not permitted	Controls maximum time after last update before default values are used
SCPTlocation applying to SFPTdischargeAirController	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTbypassTime applying to SFPTdischargeAirController	Optional	none	not permitted	Maximum time the controller can be in bypass mode following request
SCPTmaxSupplyFanCapacity applying to SFPTdischargeAirController	Optional	none	not permitted	Maximum supply fan capacity setpoint for the Discharge-Air Controller
SCPTminSupplyFanCapacity applying to SFPTdischargeAirController	Optional	none	not permitted	Minimum supply fan capacity setpoint for the Discharge-Air Controller
SCPTmaxReturnExhaustFanCapacity applying to SFPTdischargeAirController	Optional	none	not permitted	Maximum return/exhaust fan capacity setpoint for the Discharge-Air Controller
SCPTminReturnExhaustFanCapacity applying to SFPTdischargeAirController	Optional	none	not permitted	Minimum return/exhaust fan capacity setpoint for the Discharge-Air Controller
SCPTductStaticPressureSetpoint applying to SFPTdischargeAirController	Optional	none	not permitted	Default duct static pressure setpoint for the Discharge-Air Controller

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxDuctStaticPressureSetpoint applying to SFPTdischargeAirController	Optional	none	not permitted	Maximum duct static pressure setpoint for the Discharge-Air Controller
SCPTminDuctStaticPressureSetpoint applying to SFPTdischargeAirController	Optional	none	not permitted	Minimum duct static pressure setpoint for the Discharge-Air Controller
SCPTductStaticPressureLimit applying to SFPTdischargeAirController	Optional	none	not permitted	Duct static pressure limit, used for equipment protection
SCPTbuildingStaticPressureSetpoint applying to SFPTdischargeAirController	Optional	none	not permitted	Default building static pressure setpoint for the Discharge-Air Controller
SCPTreturnFanStaticPressureSetpoint applying to SFPTdischargeAirController	Optional	none	not permitted	Return fan static pressure setpoint for the Discharge-Air Controller
SCPTfanDifferentialSetpoint applying to SFPTdischargeAirController	Optional	none	not permitted	Default for percent capacity difference between supply and return fans
SCPTmixedAirLowLimitSetpoint applying to SFPTdischargeAirController	Optional	none	not permitted	Mixed air low limit setpoint for the Discharge-Air Controller
SCPTmixedAirTempSetpoint applying to SFPTdischargeAirController	Optional	none	not permitted	Default mixed air temperature setpoint for the Discharge-Air Controller
SCPTminRnge applying to nvoOADamper	Optional	none	not permitted	Default minimum outdoor air damper position setpoint for the Discharge-Air Controller

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTminOutdoorAirFlowSetpoint applying to SFPTdischargeAirController	Optional	none	not permitted	Default minimum outdoor airflow setpoint for the Discharge-Air Controller
SCPTsensConstVAV applying to SFPTdischargeAirController	Optional	none	not permitted	Gain for the outdoor airflow calibration for the Discharge-Air Controller
SCPTductArea applying to SFPTdischargeAirController	Optional	none	not permitted	Area of the outdoor air inlet for the Discharge-Air Controller
SCPToutdoorAirTempSetpoint applying to SFPTdischargeAirController	Optional	none	not permitted	Airside economizer outdoor air temperature enable setpoint for the Discharge-Air Controller
SCPToutdoorAirEnthalpySetpoint applying to SFPTdischargeAirController	Optional	none	not permitted	Default airside economizer outdoor air enthalpy enable setpoint for the Discharge-Air Controller
SCPTdiffTempSetpoint applying to SFPTdischargeAirController	Optional	none	not permitted	Differential between entering air temp and entering condenser water temp to enable economizer operation
SCPTexhaustEnablePosition applying to SFPTdischargeAirController	Optional	none	not permitted	Exhaust enable outdoor air damper position setpoint for the Discharge-Air Controller
SCPTspaceHumSetpoint applying to SFPTdischargeAirController	Optional	none	not permitted	Default space humidification setpoint for the Discharge-Air Controller
SCPThumSetpt applying to SFPTdischargeAirController	Optional	none	not permitted	Default space dehumidification setpoint for the Discharge-Air Controller

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTdischargeAirDewpointSetpoint applying to SFPTdischargeAirController	Optional	none	not permitted	Default discharge-air dewpoint setpoint for the Discharge-Air Controller
SCPTmaxDischargeAirCoolingSetpoint applying to SFPTdischargeAirController	Optional	none	not permitted	Maximum discharge-air cooling setpoint for the Discharge-Air Controller
SCPTminDischargeAirCoolingSetpoint applying to SFPTdischargeAirController	Optional	none	not permitted	Minimum discharge-air cooling setpoint for the Discharge-Air Controller
SCPTmaxDischargeAirHeatingSetpoint applying to SFPTdischargeAirController	Optional	none	not permitted	Maximum discharge heating setpoint for the Discharge-Air Controller
SCPTminDischargeAirHeatingSetpoint applying to SFPTdischargeAirController	Optional	none	not permitted	Minimum discharge heating setpoint for the Discharge-Air Controller
SCPTcoolingLockout applying to SFPTdischargeAirController	Optional	none	not permitted	Outdoor air temperature cooling lockout setpoint for the Discharge-Air Controller
SCPTheatingLockout applying to SFPTdischargeAirController	Optional	none	not permitted	Outdoor air temperature heating lockout setpoint for the Discharge-Air Controller
SCPTcoolingResetEnable applying to SFPTdischargeAirController	Optional	none	not permitted	Enables/disables the discharge-air temperature cooling reset control for the Discharge-Air Controller
SCPTheatingResetEnable applying to SFPTdischargeAirController	Optional	none	not permitted	Enables/disables the discharge-air temperature heating reset control for the Discharge-Air Controller

7.77 SFPTrailcarAudioController (9111)

NV Member	Requirement	CNP Type	Direction	Description
1 "Audio Request Input" (nviAudReq)	Mandatory	SNVT_rac_req	input	This network variable is the audio request received from the Audio Sensor Object in the car
2 "Audio Control Command Request Output" (nvoAudCntCmdReq)	Mandatory	SNVT_rac_ctrl	output	This network variable is the audio control command sent to the Audio Controller Object in the system
3 "Audio Control Command Request Input" (nviAudCntCmdReq)	Mandatory	SNVT_rac_ctrl	input	These network variables are the audio control commands received from other Audio Controller Objects in the system
4 "Audio Control Command from this Controller" (nvoAudCntCmdCont)	Mandatory	SNVT_rac_ctrl	output	This network variable is the audio control command sent to the Audio Controller Object in the system
5 "Audio Control Command from another Controller" (nviAudCntCmdCont)	Mandatory	SNVT_rac_ctrl	input	These network variables are the audio control commands received from other Audio Controller Objects in the system
6 "Audio Control Command to a Car" (nvoAudCntCmdCar)	Mandatory	SNVT_rac_ctrl	output	This network variable is the audio control command sent to the Audio Sensor functional block (nviAudioCntCmdCar) of the audio unit

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to nvoAudCntCmdReq	Mandatory	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoAudCntCmdCont	Mandatory	none	not permitted	The maximum period of time between consecutive transmissions of the current value

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to nvoAudCntCmdCar	Mandatory	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTminSendTime applying to nvoAudCntCmdReq	Mandatory	none	not permitted	The minimum period of time between consecutive transmissions of the current value
SCPTminSendTime applying to nvoAudCntCmdCont	Mandatory	none	not permitted	The minimum period of time between consecutive transmissions of the current value
SCPTminSendTime applying to nvoAudCntCmdCar	Mandatory	none	not permitted	The minimum period of time between consecutive transmissions of the current value
SCPTlocation applying to SFPTrailcarAudioController	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTobjMajVer applying to SFPTrailcarAudioController	Optional	constant	not permitted	The major version number for the object
SCPTobjMinVer applying to SFPTrailcarAudioController	Optional	device specific, constant	not permitted	The minor version number for the object

7.78 SFPTrailcarAudioSensor (9112)

NV Member	Requirement	CNP Type	Direction	Description
1 "Audio Control Command from a Car" (nviAudCntCmdCar)	Mandatory	SNVT_rac_ctrl	input	This network variable receives audio command from Audio Controller Object (nvoAudioCntCmdCar), and uses it for internal set-up
2 "Audio Request Output" (nvoAudReq)	Optional	SNVT_rac_req	output	This network variable is the audio request output which is sent to the Audio Controller Object in the car

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTlocation applying to SFPTrailcarAudioSensor	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTobjMajVer applying to SFPTrailcarAudioSensor	Optional	constant	not permitted	The major version number for the object
SCPTobjMinVer applying to SFPTrailcarAudioSensor	Optional	device specific, constant	not permitted	The minor version number for the object

7.79 SFPTrefrigDisplayCaseControllerDefrost (10010)

NV Member	Requirement	CNP Type	Direction	Description
1 "Defrost Enable" (nviDefrostEnable)	Mandatory	SNVT_lev_disc	input	Begin defrost switch
2 "Defrost state" (nvoDefrostState)	Mandatory	SNVT_defr_state	output	The nvoDefrostState indicates the current state of the defrost object.
3 "Temperture 1" (nviTemperature1)	Optional	SNVT_temp_p	input	nviTemperature1 is to be assigned to inlet or outlet of the evaporator as required. An error on the sensor is indicated with a value outside the valid range.
4 "Temperature" (nvoTemperature1)	Optional	SNVT_temp_p	output	Processed temperature 1
5 "Temperatur 2" (nviTemperture2)	Optional	SNVT_temp_p	input	The nviTemperature2 is to be assigned to inlet or outlet of the evaporator as required. An error on the sensor is indicated with a value outside the valid range.
6 "Temperature" (nvoTemperture2)	Optional	SNVT_temp_p	output	Processed temperature 2

NV Member	Requirement	CNP Type	Direction	Description
7 "Start Up" (nviStartUp)	Optional	SNVT_lev_disc	input	If synchronized defrost is selected as the control strategy, this network variable indicates that the defrost object can begin the after defrost sequence.
8 "Defrost Terminate" (nviDefTerminate)	Optional	SNVT_lev_disc	input	nviDefrostTerminate can be used as an input from some external sensor e.g. ice sensor to indicate the level of defrosting.
9 "Humidity" (nviHumidity)	Optional	SNVT_lev_percent	input	The nviHumidity can be used to provide humidity information for the defrost algorithm.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to SFPTrefrigDisplayCaseControllerDefrost	Mandatory	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTminSendTime applying to SFPTrefrigDisplayCaseControllerDefrost	Mandatory	none	not permitted	The minimum period of time between consecutive transmissions of the current value
SCPTlocation applying to SFPTrefrigDisplayCaseControllerDefrost	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTdefrostMode applying to SFPTrefrigDisplayCaseControllerDefrost	Optional	none	not permitted	The type of defrost to perform
SCPTstrtupDelay applying to SFPTrefrigDisplayCaseControllerDefrost	Optional	none	not permitted	The time to delay after power-up, defrost, or pack fail

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTtermTimeTemp applying to SFPTrefrigDisplayCaseControllerDefrost	Optional	none	not permitted	The defrost termination condition
SCPTmaxDefrstTemp applying to SFPTrefrigDisplayCaseControllerDefrost	Optional	none	not permitted	The temperature at which to terminate defrost for objects set to terminate on temperature
SCPTmaxDefrstTime applying to SFPTrefrigDisplayCaseControllerDefrost	Optional	none	not permitted	The maximum defrost time for defrost objects set to terminate on temperature
SCPTpumpDownDelay applying to SFPTrefrigDisplayCaseControllerDefrost	Optional	none	not permitted	The delay to use before starting the defrost
SCPTdrainDelay applying to SFPTrefrigDisplayCaseControllerDefrost	Optional	none	not permitted	The delay to use after the defrost has terminated
SCPTinjDelay applying to SFPTrefrigDisplayCaseControllerDefrost	Optional	none	not permitted	The delay to use after the defrost has terminated

7.80 SFPTrefrigDisplayCaseControllerEvaporator (10011)

NV Member	Requirement	CNP Type	Direction	Description
1 "Valve opening" (nvoAcuatorOpening)	Mandatory	SNVT_lev_percent	output	The current opening degree of the valve, in percent of fully open.
2 "Evaporator state" (nvoEvaporatorState)	Mandatory	SNVT_evap_state	output	The current state of the evaporator object

NV Member	Requirement	CNP Type	Direction	Description
3 "Evaporator Inlet Temperature" (nviEvapInTemp)	Optional	SNVT_temp_p	input	These values indicate the current evaporator inlet (liquid line) temperature. The input can be used if the sensor is external to the evaporator object.
4 "Evaporator Inlet Temperature" (nvoEvapInTemp)	Optional	SNVT_temp_p	output	These values indicate the current evaporator inlet (liquid line) temperature. The output can be used if the sensor is internal to the evaporator object.
5 "Evaporator Outlet Temperature" (nviEvapOutTemp)	Optional	SNVT_temp_p	input	These values indicate the current evaporator outlet (suction line) temperature. The input can be used if the sensor is external to the vaporator object.
6 "Evaporator Outlet Temperature" (nvoEvapOutTemp)	Optional	SNVT_temp_p	output	These values indicate the current evaporator outlet (suction line) temperature. The output can be used if the sensor is internal to the evaporator object.
7 "Forced Value" (nviForcedValue)	Optional	SNVT_switch	input	The nviForcedValve is used to force the valve to a given opening degree. The evaporator object will stay in this forced mode as long as SNVT_switch.state equals TRUE.
8 "Super Heat Temperature" (nvoSuperHeatTemp)	Optional	SNVT_temp_p	output	The nvoSuperHeatTemp indicates the true evaporator super heat temperature. This variable should be used only when both pressure & emperature are used for calculation. If only temperatures are used then the delta temperature output should be used.
9 "Thermostat State" (nviThermostatState)	Optional	SNVT_state	input	This NV indicates the current state of the Thermostat Object. There are currently three different control methods supported by the Thermostat Object (See PDF)
10 "Super Heat Reference Temperature" (nvoSuperHeatRef)	Optional	SNVT_temp_p	output	The nvoSuperHeatRef indicates the current target evaporator super heat temperature.

NV Member	Requirement	CNP Type	Direction	Description
11 "Defrost state" (nviDefrostState)	Optional	SNVT_defr_state	input	This NV indicates the current state of the defrost object.
12 "Calculated Air Temperature" (nviAirTemp)	Optional	SNVT_temp_p	input	This NV is the calculated case air temperature. An error on the sensor is indicated with the error value for SNVT_temp_p (0x7fff).
13 "Cut out Temperature" (nviCutoutTemp)	Optional	SNVT_temp_p	input	This NV indicates the current cut out limit used by the thermostat object in its agorithms.
14 "Difference Temperature" (nviDifference)	Optional	SNVT_temp_p	input	This NV indicates the value to be added to the nviCutoutTemp to get the thermostat cut in limit if cut in / out control is selected.
15 "Super Heat Reference Temperature" (nviSuperHeatRef)	Optional	SNVT_temp_p	input	The nviSuperHeatRef is an override input for the target super heat reference. This input should be used when the object is in override.
16 "Liquid Line Pressure" (nviPressure)	Optional	SNVT_press	input	The input variable would be included on nodes without the hardware interface to read a pressure sensor, whereas the output variable would be included on nodes with pressure sensor hardware.
17 "Delta Temperature" (nvoDeltaTemp)	Optional	SNVT_temp_p	output	This NV indicates the inferred evaporator super heat temperature. This variable should be used when pressure is not taken into account in the calculation.
18 "Liquid Line Pressure" (nvoPressure)	Optional	SNVT_press	output	The pressure of the refrigerant in the liquid (evaporator feed) line.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTlocation applying to SFPTrefrigDisplayCaseControllerEvaporator	Optional	none	not permitted	Provides descriptive physical location information related to the object.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to SFPTrefrigDisplayCaseControllerEvaporator	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTsuperHtRefMax applying to SFPTrefrigDisplayCaseControllerEvaporator	Optional	none	not permitted	Maximum value for the target super heat network variable
SCPTsuperHtRefMin applying to SFPTrefrigDisplayCaseControllerEvaporator	Optional	none	not permitted	Minimum value for the target super heat network variable
SCPTsuperHtRefInit applying to SFPTrefrigDisplayCaseControllerEvaporator	Optional	none	not permitted	Default value for the super heat target network variable
SCPTstrtupDelay applying to SFPTrefrigDisplayCaseControllerEvaporator	Optional	none	not permitted	The time to delay after power-up, defrost, or pack fail
SCPTstrtupOpen applying to SFPTrefrigDisplayCaseControllerEvaporator	Optional	none	not permitted	Maximum valve opening to use after power-up, defrost, or pack fail
SCPTrefrigGlide applying to SFPTrefrigDisplayCaseControllerEvaporator	Optional	none	not permitted	Used to characterize the glide of the refrigerant used
SCPTrefrigType applying to SFPTrefrigDisplayCaseControllerEvaporator	Optional	none	not permitted	

7.81 SFPTrefrigDisplayCaseControllerThermostat (10012)

NV Member	Requirement	CNP Type	Direction	Description
1 "Calculated Air Temperature" (nvoAirTemp)	Mandatory	SNVT_temp_p	output	This NV is the calculated case air temperature. An error on the sensor is indicated with the error value for SNVT_temp_p (0x7fff).
2 "Thermostat State" (nvoThermostatState)	Mandatory	SNVT_state	output	This NV indicates the current state of the Thermostat Object. There are currently three different control methods supported by the Thermostat Object (See PDF)
3 "Measured Air Temperature 1" (nviAirTemp1)	Optional	SNVT_temp_p	input	This NV is to be assigned to discharge or return air as required. An error on the sensor is indicated with the error value for SNVT_temp_p (0x7fff).
4 "Discrete level" (nviDayNight)	Optional	SNVT_lev_disc	input	This SNVT is obsolete. Use SNVT_switch instead.
5 "Cut out Temperature" (nvoCutoutTemp)	Optional	SNVT_temp_p	output	This NV indicates the current cut out limit used by the thermostat object in its agorithms.
6 "Measured Air Temperature 2" (nviAirTemp2)	Optional	SNVT_temp_p	input	This NV is to be assigned to discharge or return air as required. An error on the sensor is indicated with the error value for SNVT_temp_p (0x7fff).
7 "Difference Temperature" (nvoDifference)	Optional	SNVT_temp_p	output	This NV indicates the value to be added to the nviCutoutTemp to get the thermostat cut in limit if cut in / out control is selected.
8 "Defrost state" (nviDefrostState)	Optional	SNVT_defr_state	input	This NV indicates the current state of the defrost object.
9 "Alarm Air Temperature" (nvoAlarmAirTemp)	Optional	SNVT_temp_p	output	The nvoAlarmAirTemp indicates the current air temperature used by the alarm section of the thermostat object.

NV Member	Requirement	CNP Type	Direction	Description
10 "Discharge Air Temperature" (nvoDischargeTemp)	Optional	SNVT_temp_p	output	The nvoDischargeTemp indicates the current evaporator discharge air temperature used by the thermostat object.
11 "Return Air Temperature" (nvoReturnTemp)	Optional	SNVT_temp_p	output	The nvoReturnTemp indicates the current evaporator return air temperature used by the thermostat object.
12 "Actuator Valve" (nvoActuatorValve)	Optional	SNVT_lev_disc	output	This output can be used to drive a refrigeration valve or compressor.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to SFPTrefrigDisplayCaseControllerThermostat	Mandatory	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTlocation applying to SFPTrefrigDisplayCaseControllerThermostat	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTthermMode applying to SFPTrefrigDisplayCaseControllerThermostat	Optional	none	not permitted	The thermostat control strategy
SCPTcutOutValue applying to SFPTrefrigDisplayCaseControllerThermostat	Optional	none	not permitted	The cut-out limit
SCPTdiffValue applying to SFPTrefrigDisplayCaseControllerThermostat	Optional	none	not permitted	The value to be added to the cut-out value to get the cut-in limit

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTdayNightCntrl applying to SFPTrefrigDisplayCaseControllerThermostat	Optional	none	not permitted	Configures the day/night function
SCPTairTemp1Day applying to SFPTrefrigDisplayCaseControllerThermostat	Optional	none	not permitted	The air temperature weighting used during day control
SCPTairTemp1Night applying to SFPTrefrigDisplayCaseControllerThermostat	Optional	none	not permitted	The air temperature weighting used during night control
SCPTdeltaNight applying to SFPTrefrigDisplayCaseControllerThermostat	Optional	none	not permitted	The value to be added to the cut-out value to get the cut-out limit during night control
SCPTdiffNight applying to SFPTrefrigDisplayCaseControllerThermostat	Optional	none	not permitted	The value to be added to the cut-out value to get the cut-in limit during night control
SCPTairTemp1Alrm applying to SFPTrefrigDisplayCaseControllerThermostat	Optional	none	not permitted	The weighting of the air temp 1 sensor when calculating the air temp alarm
SCPThighLimTemp applying to SFPTrefrigDisplayCaseControllerThermostat	Optional	none	not permitted	The high alarm set point for the alarm air temp network variable
SCPTlowLimTemp applying to SFPTrefrigDisplayCaseControllerThermostat	Optional	none	not permitted	The low alarm set point for the alarm air temp network variable

CNP Type	Requirement	Restrictions	Arrays	Description
SCPThighLimDefrDly applying to SFPTrefrigDisplayCaseControllerThermostat	Optional	none	not permitted	The time limit before high air temp alarm during pull-down
SCPTlowLimDly applying to SFPTrefrigDisplayCaseControllerThermostat	Optional	none	not permitted	The time limit during normal operation before the alarm air temp low alarm is recognized
SCPThighLimDly applying to SFPTrefrigDisplayCaseControllerThermostat	Optional	none	not permitted	The time limit during normal operation before the alarm air temp high alarm is recognized

7.82 SFPTfireSmokeDamperActuator (11001)

NV Member	Requirement	CNP Type	Direction	Description
1 "Actuator Drive" (nviActuDrive)	Mandatory	SNVT_hvac_emerg	input	Controls the actuator position
3 "Actuator Drive Feedback" (nvoActuDriveFb)	Optional	SNVT_hvac_emerg	output	Reflects the actuator value of nviActuDrive. Used to synchronize source objects in multiple relationships.
2 "HVAC emergency mode" (nvoActuPosn)	Mandatory	SNVT_hvac_emerg	output	

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTdriveTime applying to SFPTfireSmokeDamperActuator	Mandatory	none	not permitted	The transition time for a full 100 % stroke (change from one extreme to the other)

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxRcvTime applying to nviActuDrive	Mandatory	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTdirection applying to SFPTfireSmokeDamperActuator	Mandatory	none	not permitted	The actuator sense of rotation and safety position; bit 0 set => counterclockwise, bit 1 set => damper open
SCPToffDely applying to SFPTfireSmokeDamperActuator	Mandatory	none	not permitted	The length of time that the load remains energized after a change from ON to OFF has been received
SCPTzoneNum applying to SFPTfireSmokeDamperActuator	Mandatory	none	not permitted	The zone number for the device
SCPTactuatorType applying to SFPTfireSmokeDamperActuator	Optional	none	not permitted	The identification of the exact actuator type or label
SCPTinstallDate applying to SFPTfireSmokeDamperActuator	Optional	none	not permitted	The date of installation for the device
SCPTlocation applying to SFPTfireSmokeDamperActuator	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTmaintDate applying to SFPTfireSmokeDamperActuator	Optional	none	not permitted	The date of last maintenance for the device
SCPTmanfDate applying to SFPTfireSmokeDamperActuator	Optional	manufacturer specific	not permitted	The date of manufacture for the device
SCPToemType applying to SFPTfireSmokeDamperActuator	Optional	manufacturer specific	not permitted	The label, programmed by the OEM, to identify the unit name

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to nvoActuDriveFb	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value

7.83 SFPTsmokeFireInitiatorIntelli (11002)

NV Member	Requirement	CNP Type	Direction	Description
1 "Fire alarm" (nvoFireAlm)	Mandatory	SNVT_switch	output	Fire information for use by simple indicators
2 "Fire trouble" (nvoFireTrouble)	Mandatory	SNVT_switch	output	Initiator trouble information for use by simple indicators
3 "Device request" (nviDeviceRequest)	Mandatory	SNVT_fire_test	input	Receives requests intended to perform smoke detector operations, initiated from operator
4 "Day (0 %; 0) and Night (100 %; 1) mode" (nvoDayNightMode)	Optional	SNVT_switch	output	Present value for Day/Night mode. The DayNightMode variable is used by applications that use day/night - sensitive limits depending upon time of day.
5 "Environmental compensation" (nvoEnvComp)	Optional	SNVT_lev_cont	output	For use by operator-interface devices requiring system, environmental-compensation reporting
6 "Emergency mode: EMERG_NORMAL or EMERG_FIRE" (nvoEmergMode)	Optional	SNVT_hvac_emerg	output	Controls the (actuator) position for smoke-control devices

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTlocation applying to SFPTsmokeFireInitiatorIntelli	Mandatory	modify when disabled	not permitted	Provides descriptive physical location information related to the object.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to nvoFireAlm	Mandatory	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPToemType applying to SFPTsmokeFireInitiatorIntelli	Optional	manufacturer specific, constant	not permitted	The label, programmed by the OEM, to identify the unit name
SCPTsmokeNomSens applying to SFPTsmokeFireInitiatorIntelli	Mandatory	modify when disabled	not permitted	The nominal sensitivity value for the fire initiator in percentage obscuration by smoke
SCPTsmokeDayAlrmLim applying to SFPTsmokeFireInitiatorIntelli	Mandatory	modify when disabled	not permitted	The daytime alarm limit sensitivity value for the fire initiator in percentage obscuration by smoke
SCPTsmokeNightAlrmLim applying to SFPTsmokeFireInitiatorIntelli	Mandatory	modify when disabled	not permitted	The nighttime alarm limit sensitivity value for the fire initiator in percentage obscuration by smoke
SCPTsmokeDayPreAlrmLim applying to SFPTsmokeFireInitiatorIntelli	Mandatory	modify when disabled	not permitted	The daytime pre-alarm limit sensitivity value for the fire initiator in percentage obscuration by smoke
SCPTsmokeNightPreAlrmLim applying to SFPTsmokeFireInitiatorIntelli	Mandatory	modify when disabled	not permitted	The nighttime pre-alarm limit sensitivity value for the fire initiator in percentage obscuration by smoke
SCPTzoneNum applying to SFPTsmokeFireInitiatorIntelli	Mandatory	none	not permitted	The zone number for the device
SCPTinstallDate applying to SFPTsmokeFireInitiatorIntelli	Optional	modify when disabled	not permitted	The date of installation for the device
SCPTmaintDate applying to SFPTsmokeFireInitiatorIntelli	Optional	none	not permitted	The date of last maintenance for the device

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmanfDate applying to SFPTsmokeFireInitiatorIntelli	Optional	manufacturer specific, constant	not permitted	The date of manufacture for the device
SCPTfireTxt1 applying to SFPTsmokeFireInitiatorIntelli	Optional	modify when disabled	not permitted	Text information relevant to fire conditions. A '>' at end of string indicates presence of fire text 2.
SCPTfireTxt2 applying to SFPTsmokeFireInitiatorIntelli	Optional	modify when disabled	not permitted	Continuation text information relevant to fire conditions. A '>' at end of string indicates presence of fire text 3.
SCPTfireTxt3 applying to SFPTsmokeFireInitiatorIntelli	Optional	modify when disabled	not permitted	Second continuation text information relevant to fire conditions

7.84 SFPTsmokeFireInitiatorConvent (11003)

NV Member	Requirement	CNP Type	Direction	Description
3 "Device Request" (nviDeviceRequest)	Mandatory	SNVT_fire_test	input	This input NV receives requests intended to perform smoke detector operations, initiated from operator
1 "Fire Alarm" (nvoFireAlm)	Mandatory	SNVT_switch	output	This outpur NV transmits fire information for use by simple Indicators
2 "Fire Trouble" (nvoFireTrouble)	Mandatory	SNVT_switch	output	This output NV transmitted initiator trouble information for use by simple Indicators
4 "HVAC emergency mode" (nvoEmergMode)	Optional	SNVT_hvac_emerg	output	

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTlocation applying to SFPTsmokeFireInitiatorConvent	Mandatory	modify when disabled	not permitted	Provides descriptive physical location information related to the object.
SCPTmaxSendTime applying to nviDeviceRequest	Mandatory	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTzoneNum applying to SFPTsmokeFireInitiatorConvent	Mandatory	none	not permitted	The zone number for the device
SCPToemType applying to SFPTsmokeFireInitiatorConvent	Optional	manufacturer specific, constant	not permitted	The label, programmed by the OEM, to identify the unit name
SCPTinstallDate applying to SFPTsmokeFireInitiatorConvent	Optional	modify when disabled	not permitted	The date of installation for the device
SCPTmaintDate applying to SFPTsmokeFireInitiatorConvent	Optional	none	not permitted	The date of last maintenance for the device
SCPTmanfDate applying to SFPTsmokeFireInitiatorConvent	Optional	manufacturer specific, constant	not permitted	The date of manufacture for the device
SCPTfireTxt1 applying to SFPTsmokeFireInitiatorConvent	Optional	modify when disabled	not permitted	Text information relevant to fire conditions. A '>' at end of string indicates presence of fire text 2.
SCPTfireTxt2 applying to SFPTsmokeFireInitiatorConvent	Optional	modify when disabled	not permitted	Continuation text information relevant to fire conditions. A '>' at end of string indicates presence of fire text 3.
SCPTfireTxt3 applying to SFPTsmokeFireInitiatorConvent	Optional	modify when disabled	not permitted	Second continuation text information relevant to fire conditions

7.85 SFPTthermalFireInitiator (11004)

NV Member	Requirement	CNP Type	Direction	Description
1 "Fire Alarm" (nvoFireAlm)	Mandatory	SNVT_switch	output	This outpur NV transmits fire information for use by simple Indicators
2 "Fire trouble" (nvoFireTrouble)	Mandatory	SNVT_switch	output	This output NV transmitted initiator trouble information for use by simple Indicators
3 "Fire test request" (nviDeviceRequest)	Mandatory	SNVT_fire_test	input	This input NV receives requests intended to perform smoke detector operations, initiated from operator
4 "HVAC emergency mode" (nvoEmergMode)	Optional	SNVT_hvac_emerg	output	This input NV controls the (actuator) position for smoke control devices

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTlocation applying to SFPTthermalFireInitiator	Mandatory	modify when disabled	not permitted	Provides descriptive physical location information related to the object.
SCPTmaxSendTime applying to nvoFireAlm	Mandatory	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTzoneNum applying to SFPTthermalFireInitiator	Mandatory	none	not permitted	The zone number for the device
SCPToemType applying to SFPTthermalFireInitiator	Optional	manufacturer specific, constant	not permitted	The label, programmed by the OEM, to identify the unit name
SCPTinstallDate applying to SFPTthermalFireInitiator	Optional	modify when disabled	not permitted	The date of installation for the device

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaintDate applying to SFPTthermalFireInitiator	Optional	none	not permitted	The date of last maintenance for the device
SCPTmanfDate applying to SFPTthermalFireInitiator	Optional	manufacturer specific, constant	not permitted	The date of manufacture for the device
SCPTfireTxt1 applying to SFPTthermalFireInitiator	Optional	modify when disabled	not permitted	Text information relevant to fire conditions. A '>' at end of string indicates presence of fire text 2.
SCPTfireTxt2 applying to SFPTthermalFireInitiator	Optional	modify when disabled	not permitted	Continuation text information relevant to fire conditions. A '>' at end of string indicates presence of fire text 3.
SCPTfireTxt3 applying to SFPTthermalFireInitiator	Optional	modify when disabled	not permitted	Second continuation text information relevant to fire conditions
SCPTthermAlrmROR applying to SFPTthermalFireInitiator	Optional	manufacturer specific, constant	not permitted	The thermal alarm trip rate of rise
SCPTthermThreshold applying to SFPTthermalFireInitiator	Optional	manufacturer specific, constant	not permitted	Thermal alarm trip threshold for the initiator

7.86 SFPTpullStationFireInitiator (11005)

NV Member	Requirement	CNP Type	Direction	Description
1 "Fire Alarm" (nvoFireAlm)	Mandatory	SNVT_switch	output	This outpur NV transmits fire information for use by simple Indicators
2 "Fire Trouble" (nvoFireTrouble)	Mandatory	SNVT_switch	output	This output NV transmitted initiator trouble information for use by simple Indicators

NV Member	Requirement	CNP Type	Direction	Description
3 "Fire test request" (nviDeviceRequest)	Mandatory	SNVT_fire_test	input	This input NV receives requests intended to perform smoke detector operations, initiated from operator
4 "HVAC emergency mode" (nvoEmergMode)	Optional	SNVT_hvac_emerg	output	This input NV controls the (actuator) position for smoke control devices

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTlocation applying to SFPTpullStationFireInitiator	Mandatory	modify when disabled	not permitted	Provides descriptive physical location information related to the object.
SCPTmaxSendTime applying to nvoFireAlm	Mandatory	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTzoneNum applying to SFPTpullStationFireInitiator	Mandatory	none	not permitted	The zone number for the device
SCPToemType applying to SFPTpullStationFireInitiator	Optional	manufacturer specific, constant	not permitted	The label, programmed by the OEM, to identify the unit name
SCPTinstallDate applying to SFPTpullStationFireInitiator	Optional	modify when disabled	not permitted	The date of installation for the device
SCPTmaintDate applying to SFPTpullStationFireInitiator	Optional	none	not permitted	The date of last maintenance for the device
SCPTmanfDate applying to SFPTpullStationFireInitiator	Optional	manufacturer specific, constant	not permitted	The date of manufacture for the device

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTfireTxt1 applying to SFPTpullStationFireInitiator	Optional	modify when disabled	not permitted	Text information relevant to fire conditions. A '>' at end of string indicates presence of fire text 2.
SCPTfireTxt2 applying to SFPTpullStationFireInitiator	Optional	modify when disabled	not permitted	Continuation text information relevant to fire conditions. A '>' at end of string indicates presence of fire text 3.
SCPTfireTxt3 applying to SFPTpullStationFireInitiator	Optional	modify when disabled	not permitted	Second continuation text information relevant to fire conditions

7.87 SFPTaudibleFireIndicator (11006)

NV Member	Requirement	CNP Type	Direction	Description
1 "Fire Audible Input" (nviFireAudible)	Mandatory	SNVT_switch	input	This input network variable receives the status (ON or OFF) request (command) for an audible indicating device. It can be bound to the nvoAlarm network variable(s) of initiating device(s)
2 "Switch" (nvoFireTrouble)	Mandatory	SNVT_switch	output	
3 "Fire Audible Feedback" (nvoFireAudible)	Optional	SNVT_switch	output	This output network variable transmits the feedback status (ON or OFF) of indicating device.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTlocation applying to SFPTaudibleFireIndicator	Mandatory	modify when disabled	not permitted	Provides descriptive physical location information related to the object.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxRcvTime applying to SFPTaudibleFireIndicator	Mandatory	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTzoneNum applying to SFPTaudibleFireIndicator	Mandatory	none	not permitted	The zone number for the device
SCPTaudOutput applying to SFPTaudibleFireIndicator	Mandatory	manufacturer specific, constant	not permitted	Audible sound output intensity specification of the device at 1 meter distant
SCPToemType applying to SFPTaudibleFireIndicator	Optional	manufacturer specific, constant	not permitted	The label, programmed by the OEM, to identify the unit name
SCPTinstallDate applying to SFPTaudibleFireIndicator	Optional	modify when disabled	not permitted	The date of installation for the device
SCPTmaintDate applying to SFPTaudibleFireIndicator	Optional	none	not permitted	The date of last maintenance for the device
SCPTmanfDate applying to SFPTaudibleFireIndicator	Optional	manufacturer specific, constant	not permitted	The date of manufacture for the device
SCPTfireTxt1 applying to SFPTaudibleFireIndicator	Optional	modify when disabled	not permitted	Text information relevant to fire conditions. A '>' at end of string indicates presence of fire text 2.
SCPTfireTxt2 applying to SFPTaudibleFireIndicator	Optional	modify when disabled	not permitted	Continuation text information relevant to fire conditions. A '>' at end of string indicates presence of fire text 3.
SCPTfireTxt3 applying to SFPTaudibleFireIndicator	Optional	modify when disabled	not permitted	Second continuation text information relevant to fire conditions

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to SFPTaudibleFireIndicator	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTfireIndicate applying to SFPTaudibleFireIndicator	Optional	none	not permitted	Describes the fire indicator device

7.88 SFPTvisibleFireIndicator (11007)

NV Member	Requirement	CNP Type	Direction	Description
1 "Fire Visible Input" (nviFireVisible)	Mandatory	SNVT_switch	input	This input network variable receives the status (ON or OFF) request (command) for a Visible indicating device. It can be bound to the nvoAlarm network variable(s) of initiating device(s)
2 "Fire Trouble" (nvoFireTrouble)	Mandatory	SNVT_switch	output	This output NV transmitted initiator trouble information for use by simple Indicators
3 "Fire Visible Feedback" (nvoFireVisible)	Optional	SNVT_switch	output	This output network variable transmits the feedback status (ON or OFF) of indicating evice.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTlocation applying to SFPTvisibleFireIndicator	Mandatory	modify when disabled	not permitted	Provides descriptive physical location information related to the object.
SCPTmaxRcvTime applying to SFPTvisibleFireIndicator	Mandatory	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTzoneNum applying to SFPTvisibleFireIndicator	Mandatory	none	not permitted	The zone number for the device
SCPTvisOutput applying to SFPTvisibleFireIndicator	Mandatory	device specific, constant	not permitted	Visible light output intensity specification of the device at 0 deg viewing angle
SCPTflashFreq applying to SFPTvisibleFireIndicator	Mandatory	device specific, constant	not permitted	Flash rate specification for visible indication (strobe) device
SCPTmaxSendTime applying to SFPTvisibleFireIndicator	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPToemType applying to SFPTvisibleFireIndicator	Optional	manufacturer specific, constant	not permitted	The label, programmed by the OEM, to identify the unit name
SCPTinstallDate applying to SFPTvisibleFireIndicator	Optional	none	not permitted	The date of installation for the device
SCPTmaintDate applying to SFPTvisibleFireIndicator	Optional	none	not permitted	The date of last maintenance for the device
SCPTmanfDate applying to SFPTvisibleFireIndicator	Optional	manufacturer specific, constant	not permitted	The date of manufacture for the device
SCPTfireTxt1 applying to SFPTvisibleFireIndicator	Optional	modify when disabled	not permitted	Text information relevant to fire conditions. A '>' at end of string indicates presence of fire text 2.
SCPTfireTxt2 applying to SFPTvisibleFireIndicator	Optional	modify when disabled	not permitted	Continuation text information relevant to fire conditions. A '>' at end of string indicates presence of fire text 3.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTfireTxt3 applying to SFPTvisibleFireIndicator	Optional	modify when disabled	not permitted	Second continuation text information relevant to fire conditions
SCPTfireIndicate applying to SFPTvisibleFireIndicator	Optional	none	not permitted	Describes the fire indicator device

7.89 SFPTuniversalFireInitiator (11010)

NV Member	Requirement	CNP Type	Direction	Description
1 "Universal Initiator State" (nvoUFIState)	Mandatory	SNVT_switch	output	This output NV reflects the condition of the initiator device
2 "Device Request" (nviDeviceRq)	Optional	SNVT_fire_test	input	This input NV controls the UFI object
3 "Initiator Trouble" (nvoFireTrouble)	Optional	SNVT_switch	output	This output NV reflects the operational condititon of the initiator device

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTfireInitType applying to SFPTuniversalFireInitiator	Mandatory	none	not permitted	The fire initiator type identifier, entered into the device at installation and/or configuration time
SCPTlocation applying to SFPTuniversalFireInitiator	Mandatory	modify when disabled	not permitted	Provides descriptive physical location information related to the object.
SCPTmaxSendTime applying to SFPTuniversalFireInitiator	Mandatory	none	not permitted	The maximum period of time between consecutive transmissions of the current value

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTzoneNum applying to SFPTuniversalFireInitiator	Mandatory	none	not permitted	The zone number for the device
SCPTfireTxt1 applying to SFPTuniversalFireInitiator	Optional	modify when disabled	not permitted	Text information relevant to fire conditions. A '>' at end of string indicates presence of fire text 2.
SCPTfireTxt2 applying to SFPTuniversalFireInitiator	Optional	modify when disabled	not permitted	Continuation text information relevant to fire conditions. A '>' at end of string indicates presence of fire text 3.
SCPTfireTxt3 applying to SFPTuniversalFireInitiator	Optional	modify when disabled	not permitted	Second continuation text information relevant to fire conditions
SCPTinvrtOut applying to SFPTuniversalFireInitiator	Optional	none	not permitted	This parameter indicates to invert the active polarity, if the value is nonzero (ON).
SCPTmaintDate applying to SFPTuniversalFireInitiator	Optional	none	not permitted	The date of last maintenance for the device
SCPTmanfDate applying to SFPTuniversalFireInitiator	Optional	manufacturer specific, constant	not permitted	The date of manufacture for the device
SCPToemType applying to SFPTuniversalFireInitiator	Optional	manufacturer specific, constant	not permitted	The label, programmed by the OEM, to identify the unit name

7.90 SFPTuniversalFireIndicator (11011)

NV Member	Requirement	CNP Type	Direction	Description
1 "Universal Fire Indicator State" (nviUFIState)	Mandatory	SNVT_switch	input	The input NV controls the indicator device
2 "Indicator Trouble" (nvoFireTrouble)	Optional	SNVT_switch	output	This output NV reflects the operational condition of the indicator device

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTfireIndicate applying to SFPTuniversalFireIndicator	Mandatory	none	not permitted	Describes the fire indicator device
SCPTlocation applying to SFPTuniversalFireIndicator	Mandatory	none	not permitted	Provides descriptive physical location information related to the object.
SCPTmaxRcvTime applying to SFPTuniversalFireIndicator	Mandatory	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTzoneNum applying to SFPTuniversalFireIndicator	Mandatory	none	not permitted	The zone number for the device
SCPTfireTxt1 applying to SFPTuniversalFireIndicator	Optional	modify when disabled	not permitted	Text information relevant to fire conditions. A '>' at end of string indicates presence of fire text 2.
SCPTfireTxt2 applying to SFPTuniversalFireIndicator	Optional	modify when disabled	not permitted	Continuation text information relevant to fire conditions. A '>' at end of string indicates presence of fire text 3.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTfireTxt3 applying to SFPTuniversalFireIndicator	Optional	modify when disabled	not permitted	Second continuation text information relevant to fire conditions
SCPTinstallDate applying to SFPTuniversalFireIndicator	Optional	none	not permitted	The date of installation for the device
SCPTmaintDate applying to SFPTuniversalFireIndicator	Optional	none	not permitted	The date of last maintenance for the device
SCPTmanfDate applying to SFPTuniversalFireIndicator	Optional	manufacturer specific, constant	not permitted	The date of manufacture for the device
SCPToemType applying to SFPTuniversalFireIndicator	Optional	manufacturer specific, constant	not permitted	The label, programmed by the OEM, to identify the unit name

7.91 SFPTgeneratorSet (13110)

NV Member	Requirement	CNP Type	Direction	Description
1 "Start Command Input" (nviStartCmd)	Mandatory	SNVT_switch	input	start and stop the generator set
2 "Fault Reset Command Input" (nviFaultResetCmd)	Mandatory	SNVT_switch	input	reset or clear a generator set fault
3 "Run Status Output" (nvoRunStatus)	Mandatory	SNVT_switch	output	running at rated speed and voltage and is ready to accept load
4 "Fault Status Output" (nvoFaultStatus)	Mandatory	SNVT_switch	output	report the presence of a generator set fault

NV Member	Requirement	CNP Type	Direction	Description
5 "Shutdown Command Input" (nviShutdownCmd)	Optional	SNVT_switch	input	emergency, unconditional shutdown/disable of run
6 "NFPA 110 Annunciation Output" (nvoNFPA110Annun)	Optional	SNVT_state	output	report state of National Fire Protection Agency (USA) genset faults (NFPA §110)
7 "Frequency Output" (nvoFrequency)	Optional	SNVT_freq_hz	output	output frequency of the generator set
8 "Line-to-Line Voltage Output" (nvoVoltageLL)	Optional	SNVT_volt_ac	output	line-to-line output voltage(s) of the generator set
9 "Line-to-Neutral Voltage Output" (nvoVoltageLN)	Optional	SNVT_volt_ac	output	line-to-neutral output voltage(s) of the generator set
10 "Line Current Output" (nvoCurrent)	Optional	SNVT_amp_ac	output	output line current(s) of the generator set
11 "Power Factor Output" (nvoPowerFactor)	Optional	SNVT_pwr_fact	output	power factor of the generator set
12 "Real Power Output" (nvoRealPower)	Optional	SNVT_power_f	output	real power output as a floating type (in Watts)
13 "Generated Energy Output" (nvoGenEnergy)	Optional	SNVT_elec_whr_f	output	total (cumulative) electrical energy (WHR) generated by genset
14 "Engine Speed Output" (nvoEngineSpeed)	Optional	SNVT_rpm	output	engine speed of the generator set

NV Member	Requirement	CNP Type	Direction	Description
15 "Engine Temperature Output" (nvoEngineTemp)	Optional	SNVT_temp	output	engine temperature of the generator set
16 "Engine Oil Pressure Output" (nvoOilPressure)	Optional	SNVT_press	output	engine oil pressure of the generator set
17 "Battery Voltage Output" (nvoBattery)	Optional	SNVT_volt	output	starting battery voltage of the engine
18 "Engine Starts Output" (nvoEngineStarts)	Optional	SNVT_count	output	total number of successful engine starts
19 "Engine Run Time Output" (nvoEngineRunTime)	Optional	SNVT_time_f	output	total (cumulative) run time of the engine

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTlocation applying to SFPTgeneratorSet	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTmaxSendTime applying to SFPTgeneratorSet	Mandatory	none	permitted with dimension 0..16	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoRunStatus	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoGenEnergy	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoEngineSpeed	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to nvoEngineTemp	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoOilPressure	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoBattery	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoEngineRunTime	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value

7.92 SFPTautomaticTransferSwitch (13120)

NV Member	Requirement	CNP Type	Direction	Description
1 "Test Command Input" (nviTestCmd)	Mandatory	SNVT_switch	input	Allows an external node to test the ATS operation
2 "Start Source 2 Command Output" (nvoSrc2StartCmd)	Mandatory	SNVT_switch	output	Allows the ATS to start and stop the alternate or emergency power source
3 "Source 1 Available Output" (nvoSrc1Available)	Mandatory	SNVT_switch	output	Indicates the availability of the normal power source (GenSet/Source 1)
4 "Source 2 Available Output" (nvoSrc2Available)	Mandatory	SNVT_switch	output	Indicates the availability of the emergency power source (GenSet/Source 2)
5 "Source 1 Connected Output" (nvoSrc1Connected)	Mandatory	SNVT_switch	output	Indicates the switch position of the ATS

NV Member	Requirement	CNP Type	Direction	Description
6 "Source 2 Connected Output" (nvoSrc2Connected)	Mandatory	SNVT_switch	output	Indicates the switch position of the ATS
7 "Fault Reset Command Input" (nviFaultResetCmd)	Optional	SNVT_switch	input	Resets or attempts to clear an ATS fault
8 "Load Shed Command Input" (nviLoadShedCmd)	Optional	SNVT_switch	input	Sheds the electrical load of the ATS
9 "Transfer Inhibit Command Input" (nviTransInhCmd)	Optional	SNVT_switch	input	Inhibits the ATS from automatically transferring the load to the emergency power source (Source 2), when Source 2 becomes available
10 "Retransfer Inhibit Command Input" (nviRetransInhCmd)	Optional	SNVT_switch	input	Inhibits the ATS from automatically transferring the load back to the normal power source (Source 1), when Source 1 becomes available
11 "Override Command Input" (nviOverrideCmd)	Optional	SNVT_switch	input	Overrides a transfer or retransfer time delay, or transfer or retransfer inhibit
12 "Start Source 1 Command Output" (nvoSrc1StartCmd)	Optional	SNVT_switch	output	Allows the ATS to start and stop the Source 1
13 "Fault Status Output" (nvoFaultStatus)	Optional	SNVT_switch	output	Reports the presence of an ATS fault
14 "Source 1 Frequency Output" (nvoSrc1Frequency)	Optional	SNVT_freq_hz	output	Line frequency of Source 1
15 "Source 2 Frequency Output" (nvoSrc2Frequency)	Optional	SNVT_freq_hz	output	Line frequency of Source 2

NV Member	Requirement	CNP Type	Direction	Description
16 "Source 1 Line-to-Line Voltage Output" (nvoSrc1VoltageLL)	Optional	SNVT_volt_ac	output	Line-to-line voltage(s) of Source 1
17 "Source 2 Line-to-Line Voltage Output" (nvoSrc2VoltageLL)	Optional	SNVT_volt_ac	output	Line-to-line voltage(s) of Source 2
18 "Source 1 Line-to-Neutral Voltage Output" (nvoSrc1VoltageLN)	Optional	SNVT_volt_ac	output	Line-to-neutral voltage(s) of Source 1
19 "Source 2 Line-to-Neutral Voltage Output" (nvoSrc2VoltageLN)	Optional	SNVT_volt_ac	output	Line-to-neutral voltage(s) of Source 2
20 "Load Current Output" (nvoLoadCurrent)	Optional	SNVT_amp_ac	output	Line current(s) of the electrical load
21 "Load Power Factor Output" (nvoLoadPF)	Optional	SNVT_pwr_fact	output	Power factor of the electrical load
22 "Load Real Power Output" (nvoLoadRealPower)	Optional	SNVT_power_f	output	Real power of the electrical load as a floating type (in Watts)

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTlocation applying to SFPTautomaticTransferSwitch	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTmaxSendTime applying to nvoSrc1Frequency	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxSendTime applying to nvoSrc2Frequency	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoSrc1VoltageLL	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoSrc2VoltageLL	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoSrc1VoltageLN	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoSrc2VoltageLN	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoLoadCurrent	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoLoadPF	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to nvoLoadRealPower	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTmaxSendTime applying to SFPTautomaticTransferSwitch	Optional	none	permitted with dimension 0..16	The maximum period of time between consecutive transmissions of the current value

7.93 SFPTelevatorPositionIndicator (14011)

NV Member	Requirement	CNP Type	Direction	Description
1 "Floor name" (nviFloorName)	Mandatory	SNVT_str_asc	input	Chosen text for floor name
2 "Message text, string 1" (nviMessageText1)	Optional	SNVT_str_asc	input	Scrolling message - characters 1-29
3 "Message text, string 2" (nviMessageText2)	Optional	SNVT_str_asc	input	Scrolling message - characters 30-58
4 "Message text, string 3" (nviMessageText3)	Optional	SNVT_str_asc	input	Scrolling message - characters 59-87
5 "Message text, string 4" (nviMessageText4)	Optional	SNVT_str_asc	input	Scrolling message - characters 88-116
6 "Car-up direction" (nviCarUp)	Optional	SNVT_switch	input	Car traveling in up direction
7 "Car-down direction" (nviCarDown)	Optional	SNVT_switch	input	Car traveling in down direction
8 "Car position" (nviCarPosition)	Optional	SNVT_count	input	Position of car to nearest floor, starting with 1. 0 means car position unknown.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxRcvTime applying to SFPTelevatorPositionIndicator	Mandatory	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTlocation applying to SFPTelevatorPositionIndicator	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTobjMajVer applying to SFPTelevatorPositionIndicator	Optional	constant	not permitted	The major version number for the object
SCPTobjMinVer applying to SFPTelevatorPositionIndicator	Optional	device specific, constant	not permitted	The minor version number for the object
SCPTnwrkCnfg applying to SFPTelevatorPositionIndicator	Optional	none	not permitted	The value of this field determines the source of the node's network configuration.
SCPTbrightness applying to SFPTelevatorPositionIndicator	Optional	none	not permitted	The brightness output of a display device
SCPTorientation applying to SFPTelevatorPositionIndicator	Optional	none	not permitted	The orientation angle of the display image (0 = landscape, 90 = portrait)
SCPTscrollSpeed applying to SFPTelevatorPositionIndicator	Optional	none	not permitted	The scroll speed of the display image

7.94 SFPTelevatorHallLantern (14012)

NV Member	Requirement	CNP Type	Direction	Description
1 "Up direction signal" (nviUpHall)	Mandatory	SNVT_switch	input	Instructs the lantern to display an 'UP' indication
3 "Location of elevator car" (nviFloorLevel)	Optional	SNVT_count	input	Location of the elevator car -- used for the scenario where all the lanterns are bound to the same direction output-network-variables

NV Member	Requirement	CNP Type	Direction	Description
2 "Down direction signal" (nviDownHall)	Mandatory	SNVT_switch	input	Instructs the lantern to display a 'DOWN' indication

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxRcvTime applying to SFPTelevatorHallLantern	Mandatory	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables.
SCPTlocation applying to SFPTelevatorHallLantern	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTobjMajVer applying to SFPTelevatorHallLantern	Optional	constant	not permitted	The major version number for the object
SCPTobjMinVer applying to SFPTelevatorHallLantern	Optional	device specific, constant	not permitted	The minor version number for the object
SCPTnwrkCnfg applying to SFPTelevatorHallLantern	Optional	none	not permitted	The value of this field determines the source of the node's network configuration.
SCPTbrightness applying to SFPTelevatorHallLantern	Optional	none	not permitted	The brightness output of a display device
SCPTinstalledLevel applying to SFPTelevatorHallLantern	Optional	none	not permitted	The floor or level on which the device is installed
SCPTscrollSpeed applying to SFPTelevatorHallLantern	Optional	none	not permitted	The scroll speed of the display image

7.95 SFPTelevatorArrivalGong (14013)

NV Member	Requirement	CNP Type	Direction	Description
1 "Up direction signal" (nviUpGong)	Mandatory	SNVT_switch	input	Instructs the gong to sound an 'UP' indication
3 "Location of elevator car" (nviFloorLevel)	Optional	SNVT_count	input	Location of the elevator car -- used for the scenario where all the gongs are bound to the same direction output-network-variables
2 "Down direction signal" (nviDownGong)	Mandatory	SNVT_switch	input	Instructs the gong to sound a 'DOWN' indication

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxRcvTime applying to SFPTelevatorArrivalGong	Mandatory	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTlocation applying to SFPTelevatorArrivalGong	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTobjMajVer applying to SFPTelevatorArrivalGong	Optional	constant	not permitted	The major version number for the object
SCPTobjMinVer applying to SFPTelevatorArrivalGong	Optional	device specific, constant	not permitted	The minor version number for the object
SCPTnwrkCnfg applying to SFPTelevatorArrivalGong	Optional	none	not permitted	The value of this field determines the source of the node's network configuration.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTinstalledLevel applying to SFPTelevatorArrivalGong	Optional	none	not permitted	The floor or level on which the device is installed

7.96 SFPTelevatorDirectionLantern (14014)

NV Member	Requirement	CNP Type	Direction	Description
1 "Up direction signal" (nviUpCar)	Mandatory	SNVT_switch	input	To instruct the lantern to display an 'UP' indication
2 "Down direction signal" (nviDownCar)	Mandatory	SNVT_switch	input	To instruct the lantern to display a 'DOWN' indication

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxRcvTime applying to SFPTelevatorDirectionLantern	Mandatory	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTlocation applying to SFPTelevatorDirectionLantern	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTobjMajVer applying to SFPTelevatorDirectionLantern	Optional	constant	not permitted	The major version number for the object
SCPTobjMinVer applying to SFPTelevatorDirectionLantern	Optional	device specific, constant	not permitted	The minor version number for the object
SCPTnwrkCnfg applying to SFPTelevatorDirectionLantern	Optional	none	not permitted	The value of this field determines the source of the node's network configuration.

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTbrightness applying to SFPTelevatorDirectionLantern	Optional	none	not permitted	The brightness output of a display device
SCPTscrollSpeed applying to SFPTelevatorDirectionLantern	Optional	none	not permitted	The scroll speed of the display image

7.97 SFPTelevatorFireSystemsPort (14041)

NV Member	Requirement	CNP Type	Direction	Description
1 "Primary fire-sensor input" (nviCommonAlarm)	Mandatory	SNVT_switch	input	Master fire signal to the elevator group controller
2 "Fire-in-lobby sensor" (nviLobbyAlarm)	Mandatory	SNVT_switch	input	Reports fire in lobby, so alternate elevator action can be taken
3 "Machine-room smoke sensor" (nviMachRoomSmoke)	Optional	SNVT_switch	input	Indicates smoke or fire in the elevator machine room
4 "Optional sensor 1" (nviOption1)	Optional	SNVT_switch	input	Optional input for installation/manufacturer-specific fire/smoke alarming connections
5 "Optional sensor 2" (nviOption2)	Optional	SNVT_switch	input	Optional input for installation/manufacturer-specific fire/smoke alarming connections
6 "Elevator is unable to respond to fire sensors" (nvoTrouble)	Optional	SNVT_switch	output	value>0 and state=1 indicates trouble

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTmaxRcvTime applying to SFPTelevatorFireSystemsPort	Mandatory	none	not permitted	The maximum period of time that may expire with no updates on the associated input network variables before the object goes into heartbeat failure mode. A zero value disables
SCPTlocation applying to SFPTelevatorFireSystemsPort	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTobjMajVer applying to SFPTelevatorFireSystemsPort	Optional	constant	not permitted	The major version number for the object
SCPTobjMinVer applying to SFPTelevatorFireSystemsPort	Optional	device specific, constant	not permitted	The minor version number for the object
SCPTmaxSendTime applying to nvoTrouble	Optional	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTzoneNum applying to SFPTelevatorFireSystemsPort	Optional	none	not permitted	The zone number for the device

7.98 SFPTelevatorVoiceAnnouncer (14061)

NV Member	Requirement	CNP Type	Direction	Description
1 "Announcement number" (nviAnnounce)	Mandatory	SNVT_count	input	External device request of announcement of a particular phrase
2 "Append a 'going down' announcement to a particular floor announcement" (nviCarDown)	Optional	SNVT_switch	input	value>0 state=1 indicates car is moving down

NV Member	Requirement	CNP Type	Direction	Description
3 "Append a 'going up' announcement to a particular floor announcement" (nviCarUp)	Optional	SNVT_switch	input	value>0 state=1 indicates car is moving up

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTlocation applying to SFPTelevatorVoiceAnnouncer	Optional	none	not permitted	Provides descriptive physical location information related to the object.
SCPTnwrkCnfg applying to SFPTelevatorVoiceAnnouncer	Optional	none	not permitted	The value of this field determines the source of the node's network configuration.
SCPTobjMajVer applying to SFPTelevatorVoiceAnnouncer	Optional	constant	not permitted	The major version number for the object
SCPTobjMinVer applying to SFPTelevatorVoiceAnnouncer	Optional	device specific, constant	not permitted	The minor version number for the object
SCPTaudibleLevel applying to nviAnnounce	Optional	none	not permitted	The audible level of announcement
SCPTaudibleLevel applying to SFPTelevatorVoiceAnnouncer	Optional	none	not permitted	The audible level of direction

7.99 SFPTclothesWasherDomestic (15011)

NV Member	Requirement	CNP Type	Direction	Description
1 "Washer Command Input" (nviWasherCmd)	Mandatory	SNVT_clothes_w_c	input	Contains all required information for clothes-washer control
2 "Washer Status Output" (nvoWasherStatus)	Mandatory	SNVT_clothes_w_s	output	Contains the present status of a washer
3 "Power On/Off Input" (nviPowerOnOff)	Mandatory	SNVT_switch	input	Commands a washer to turn-on or turn-off
4 "Power On/Off Report Output" (nvoPowerOnOffAct)	Mandatory	SNVT_switch	output	Commands a washer to turn-on or turn-off
5 "Washing Start/Stop Input" (nviStartStop)	Mandatory	SNVT_switch	input	Commands the washer to begin or halt the programmed/commanded cycle
6 "Washing Start/Stop Report Output" (nvoStartStopAct)	Mandatory	SNVT_switch	output	Reports the washer's actual state of operation
7 "Total Time Remaining Output" (nvoTimeTotalRem)	Optional	SNVT_time_min	output	This output network variable provides the total remaining time before the entire wash process is complete
8 "Alarm-Status Report Output" (nvoAlarmRpt)	Optional	SNVT_clothes_w_a	output	This output network variable provides duplicate information of the alarm field of nvoWasherStatus
9 "Washer-Management Status Output" (nvoManageStatus)	Optional	SNVT_clothes_w_m	output	This output network variable provides the status of the door/lid and the drain

CNP Type	Requirement	Restrictions	Arrays	Description
SCPTahamApplianceModel applying to SFPTclothesWasherDomestic	Mandatory	constant	not permitted	Appliance Model code as defined by the Association of Home Appliance Manufacturers
SCPTlocation applying to SFPTclothesWasherDomestic	Mandatory	none	not permitted	Provides descriptive physical location information related to the object.
SCPTmaxSendTime applying to SFPTclothesWasherDomestic	Mandatory	none	not permitted	The maximum period of time between consecutive transmissions of the current value
SCPTobjMajVer applying to SFPTclothesWasherDomestic	Optional	constant	not permitted	The major version number for the object
SCPTobjMinVer applying to SFPTclothesWasherDomestic	Optional	device specific, constant	not permitted	The minor version number for the object

8 Device-interface files

8.1 Introduction

Device-interface (XIF) files are files that define the network-visible interface for one or more CNP devices. The device interface is the interface to a device that is exposed over a CNP network. The device interface does not expose the internal algorithms of a device. Instead, it only exposes the inputs to the algorithms and the outputs from the algorithms. The device interface includes the device's self-documentation information; the number of address-table entries; the number of message tags; and the count, types, and directions of network variables.

Much of the device interface can be queried over the network by a network tool. The device manufacturer determines the completeness of a queried interface. For example, a device manufacturer may choose to embed network-variable names in a device to ensure that the queried network interface includes these names.

There are two benefits to using device-interface files. First, a device-interface file may include information that is not included in a device, such as network-variable names. Second, a device-interface file can be used during network engineering when the device is not accessible from the network-engineering tool.

The primary device-interface file type is a text file with a .xif extension.

Device-interface files are typically generated by CNP-development tools. Many of the fields of the device-interface file for a device are derived from the application in the device. If a device-interface file is modified in such a way that it does not match the application it is documenting, installation errors may occur for the device.

8.2 Text Device-Interface File Format

8.2.1 General

A text device-interface file consists of the following sections:

— Header;

— Network-variable and message-tag definitions;

— File definitions;

— Network-variable value definitions.

All sections are optional, except for the header section. If optional sections are included, those sections shall be in the specified order, as below defined. Following are a few general rules that apply to all sections:

— If the first non-blank character on a line is "#", the entire line is considered to be a comment and thus ignored by parsing tools. This means that comment lines may be inserted anywhere, since they do not count as blank lines.

— Multiple blank lines are allowed anywhere a single blank line is required, blank lines may appear between individual network-variable or message-tag records and at the end of the file, and blank characters are allowed at the beginning of any line.

— In general, string fields contain an asterisk if they are not applicable or they are default values. Integer fields contain zero when they are not applicable and asterisks when they are default values.

— The maximum line length for any line is 160 characters.

8.2.2 Header Section

8.2.2.1 General

The header section is the first section of the device-interface file, and is the only required section. The header describes some basic information about the capabilities of the device, such as the transceiver type and buffer configuration.

Installation tools may use the transceiver-type information to determine if a device is compatible with its intended channel. This usage is optional. An installation tool may use the device-interface file solely for program definition and may ignore the transceiver-type information.

Following is an example of a header section. The lines are numbered only for reference in this document; these line numbers shall not be included in the device-interface file.

1: File: EXAMPLE.XIF generated by ExampleDevelopmentTool, XIF Version 4.401

2: In conformance with EN 14908-6

3: Optional Copyright Information. Run on Tue January 01 01:23:45 2008

4:

5: 90:00:00:12:34:AB:04:02

6: 2 15 0 21 0 3 3 3 3 3 4 11 11 9 9 0 0 9 14 1 1 128 21 0 0 0 0 0 0 0 0 0 2 15 0 0 0 0 0 2 1 0

7: 0 5 100 13 28 726 0 15 5 3 342 4 10000000

8: 1 7 1 0 4 4 4 15 200 0

9: 78125 0 0 0 0 0 0 0 0 0 0 0

10: 90 0 240 0 0 0 40 40 0 5 8 5 12 14 15

11: *

12: "&3.4@0Node Object,3[2OLA,20001[2PID Controller,20010[1]Digi

13: "tal Encoder,20006[2]Analog Section,20100[2]Language Transl

14: "ator;Additional, optional descriptive text that serves an

15: "y purpose deemed appropriate by developer.

16:

The header section consists of the following lines (the Version column identifies the minimum XIF-format version required to support the entry):

Line	Version	Contents
Line 1	4.401	File name, source of the file, and format version number. This document describes format version 4.401. The format of the string shall be as follows: File: *fileName* generated by *toolName*, XIF Version *majorNum.minorNum* If the file was manually generated, specify the toolName as Manual 0.0.0. For version 4.401, specify the majorNum.minorNum as 4.401.

Line	Version	Contents
Line 2	4.401	Copyright information and/or the text "In conformance with EN 14908-6"
Line 3	4.401	Optional additional copyright information plus a required timestamp of when the file was created. The format of the string shall be as follows: *optionalInfo* Run on *day month date hour*:*min*:*sec year*
Line 4	4.401	Blank line.
Line 5	4.401	Program ID. This consists of eight 2-digit hex values, separated by colons (no spaces). The first, left-most hex digit identifies the program ID format, and shall be an '8' or '9'. The format of the program ID is the following: *FM*:*MM*:*MM*:*CC*:*CC*:*UU*:*TT*:*NN* The values are described in Annex B. The formats of the fields are described in EN 14908-5, *Standard Program ID*.
Line 6	4.401	Contains the following fields:
	4.401	Field 1 — Number of non-ECS domains. Shall be set to 2. For ECS devices, set line 6 field 33 below to the actual number of domains. May be set to 1 for devices that are not tested for compliance to EN 14908-5.
	4.401	Field 2 — Number of non-ECS address table entries. Set to 0 to 15 for non-ECS devices; for ECS devices, set to the actual number of address table entries or 15 (whichever is less) and set line 6 field 34 below to the actual number of address table entries.
	4.401	Field 3 — Boolean that specifies whether the application handles incoming application messages. Set to 1 if the application handles incoming application messages, otherwise set to 0.
	4.401	Field 4 — Number of static network variable declarations in the application. Network variables arrays count as one declaration even though each array element counts as one network variable. Set to 0 to 4096.
	4.401	Field 5 — Number of non-ECS message tags. Set to 0 to 15 for non-ECS devices; for ECS devices, set to the actual number of message tags or 15 (whichever is less) and set line 6 field 35 below to the actual number of message tags.
	4.401	Field 6 — Number of network input buffers. Encoded as follows:

Count	Encoded Value
0	0
1	2
2	3
3	4
5	5

Line	Version	Contents		
			7	6
			11	7
			15	8
			23	9
			31	10
			47	11
			63	12
			95	13
			127	14
			191	15
	4.401	Field 7	Number of network-output buffers. Encoded as described under network input buffers (field 6).	
	4.401	Field 8	Number of priority network-output buffers. Encoded as described under network-input buffers (field 6).	
	4.401	Field 9	Number of priority application output buffers. Encoded as described under network input buffers (field 6).	
	4.401	Field 10	Number of application-output buffers. Encoded as described under network-input buffers (field 6).	
	4.401	Field 11	Number of application-input buffers. Encoded as described under network-input buffers (field 6).	
	4.401	Field 12	Network-input buffer size. Encoded as follows:	
			Size	Encoded Value
			20	2
			21	3
			22	4
			24	5
			26	6
			30	7
			34	8
			42	9

Line	Version	Contents		
			50	10
			66	11
			82	12
			114	13
			146	14
			210	15
			255	0
	4.401	Field 13	Network-output buffer size. Encoded as described under network-input buffer size (field 12).	
	4.401	Field 14	Application-output buffer size. Encoded as described under network-input buffer size (field 12).	
	4.401	Field 15	Application-input buffer size. Encoded as described under network-input buffer size (field 12).	
	4.401	Field 16	Application type, encoded as follows.	
			Value	Type
			0	Unknown
			1	Network interface without a layer-7 application; no network variables or message tags
			2	Application hosted on a model 0 – 2 or 8 – 17 protocol processor
			3	Host application with host selection of network variables
			4	Host application with network-interface selection of network variables
	4.401	Field 17	Size of the network-variable configuration table for network interface applications (not host applications) with network-interface selection enabled. Set to 0 for all other applications, including host applications, ECS-device applications, and non-ECS-device applications.	
	4.401	Field 18	Number of receive transaction buffers.	
	4.401	Field 19	Number of network variable alias table entries provided by the device.	
	4.401	Field 20	Boolean that specifies whether relaxed binding constraints are allowed. If 0, each output network variable shall use a unique network variable selector. If 1, multiple output network variables can share the same selector, as long as they are not polled by an input network variable.	

Line	Version	Contents
		For non-ECS devices, set to 1 if two output network variables on the same device that are not polled by an input network variable can use the same network variable selector, otherwise set to 0. For non-ECS host-based applications using host selection, this should in general be set to 0. You can set an application to use host selection by writing the value 3 to field 16, described above. For ECS devices, set this field to 1 and then set the binding constraint number field below (field 26) to match the constraints of the device. For non-ECS-device applications, and host-based applications using network-interface selection, this should be set to match the capabilities of the protocol stack.
4.401	Field 21	Specifies whether the statistics-relative address references are allowed. Set to 1.
4.401	Field 22	Maximum size memory block that may be written at a time. For devices with flash memory, this is the flash sector size. For other devices, this value is 11 bytes.
4.401	Field 23	Maximum number of network variables this device supports, which is equal to the number of static network variables defined in field 4 plus the maximum number of dynamic network variables supported by the application. This can be no greater than 4096 and shall be greater than or equal to the number of static network variables given in field 4.
4.401	Field 24	Minimum network-management protocol version number. Set to 0.
4.401	Field 25	Maximum network-management protocol version number. Set to 1 for devices that support ECS commands. Otherwise, set to 0.
4.401	Field 26	Binding-constraint level. Set to 2 if two output network variables on the same device that are not polled by an input network variable may use the same network variable selector (in this case, field 20 should be set to 1). Otherwise, set to 1 (in this case, field 20 is set to 0).
4.401	Field 27	ECS flag 0. Set to 0 for non-ECS devices. Set to the encoded decimal value of the following bits for ECS devices: Bit — Flag Description 0 (0x01) — Fixed static NV flag. Do not set this bit if the name, self-documentation string, and rate estimates of static NVs are configurable via the UPDATE_NV_INFO ECS command. Set this bit if the name, self-documentation string, and rate estimates of static NVs are not configurable. 1 (0x02) — Incoming group restricted flag. Set this bit if incoming groups are restricted to the non-ECS address table entries. 2 – 6 — Bits 2 through 6 are reserved. Set to 0.

Line	Version	Contents
		7 (0x08) Non-unique dynamic NV names flag. Set to 0 for a device that does not support dynamic network variables and on a device that supports dynamic network variables but requires their names to be unique on the device; set to 1 for a device that supports dynamic network variables and also supports dynamic network variables with duplicate names. When creating a dynamic network variable on a device that supports duplicate dynamic network variable names, a network management tool shall ensure that the name is unique within the functional block containing the network variable, including all the static and dynamic network variables within the functional block. When creating a dynamic network variable that is not a member of a functional block, the network management tool shall ensure that the name is unique for all the static and dynamic network variables that are not members of functional blocks. Network management tools may restrict or prevent the generation of duplicate dynamic NV names.
4.401	Field 28	ECS flag 1. Set to 0 for non-ECS devices. Set to the encoded decimal value of the following bits for ECS devices: Bit — Flag Description 0 (0x01) Suppress dynamic NV definition flag. Set to 0 for a device that does not support dynamic network variables; and on a device that supports dynamic network variables and also supports the EN 14908-1 network-management commands to define dynamic network variables. Set to 1 for a device that supports dynamic network variables and does not support the EN 14908-1 network-management commands to define dynamic network variables. A device that supports dynamic network variables shall also specify a value in field 23. Set to 0 for devices tested for compliance to EN 14908-1. 1 (0x02) Suppress dynamic functional block definition flag. Set to 0 for a device that does not support dynamic functional blocks; set to 1 for a device that supports dynamic functional blocks. A device that supports dynamic functional blocks shall also specify a value in field 43. 2 (0x04) Suppress dynamic functional block member definition flag. Set to 0 for a device that does not support dynamic functional blocks; set to 1 for a device that supports dynamic functional blocks. A device that supports dynamic functional blocks shall also specify a value in field 43. 3 (0x08) Dynamic NVs supported on static functional blocks flag. Set to 0 for a device that does not support dynamic functional blocks and on a device that supports dynamic functional blocks but does not support adding dynamic network variables to static functional blocks; set to 1 for a device that supports dynamic functional blocks and also supports adding dynamic network variables to static functional blocks. A device that supports dynamic functional blocks shall also specify a value in field 43. 4 – 7 Bits 4 through 7 are reserved. Set to 0.

Line	Version	Contents	
	4.401	Fields 29–32	Reserved. Set to 0.
	4.401	Field 33	Number of domains. Set this to the value in field 1.
	4.401	Field 34	Number of address table entries. For non-ECS devices, set this to the value in field 2.
	4.401	Field 35	Number of message tags. For non-ECS devices, set this to the value in field 5.
	4.401	Field 36	Reserved. Set to 0.
	4.401	Field 37	Reserved. Set to 0.
	4.401	Field 38	Reserved. Set to 0.
	4.401	Field 39	Reserved. Set to 0.
	4.401	Field 40	The network-management version number of the device. Set to 1 for a model 0 – 2 or 8 – 17 protocol processor (as defined in Annex A) with firmware version number 13 or lower. Set to 2 for all other devices.
	4.401	Field 41	The network-management capabilities of the device. Set to 0 for a model 0 – 2 or 8 – 17 protocol processor (as defined in Annex A) with firmware version number 13 or lower. Set to 1 for all other devices.
	4.401	Field 42	Reserved. Set to 0.
	4.401	Field 43	The number of dynamic functional blocks supported by the device. Set to 0 if dynamic functional blocks are not supported. Devices that support dynamic functional blocks shall also specify a value in field 28.
Line 7		Describes the protocol-processor configuration. Line 7 contains the fields described below. Set fields 1 – 12 to 0, and set field 13 to 10000000, for host-based devices where the network image is not downloadable.	
	4.401	Field 1	Protocol processor model. Encoded as shown in Annex A.
	4.401	Field 2	Protocol processor clock rate. The value of this field will be used in conjunction with the base clock rate factor (field 13) to determine the base clock rate of the device. Encoded as follows:

Value	Rate
1	625 kHz
2	1,25 MHz
3	2,5 MHz
4	5 MHz
5	10 MHz

Line	Version	Contents	
		6	20 MHz
		7	40 MHz
	4.401	Field 3	System firmware major revision number encoded as a decimal integer value.
	4.401	Field 4	Receive transaction block size in bytes.
	4.401	Field 5	Transaction control block size in bytes.
	4.401	Field 6	Number of bytes of on-chip RAM from the end of the system area that precedes the receive transaction blocks to the first user variable or the end of on-chip RAM, whichever comes first.
	4.401	Field 7	Number of bytes of off-chip RAM from the end of the available RAM to the first user variable or the end of off-chip RAM, whichever comes first.
	4.401	Field 8	Domain-table entry size in bytes.
	4.401	Field 9	Address-table entry size in bytes.
	4.401	Field 10	Network-variable configuration-table entry size in bytes.
	4.401	Field 11	Number of bytes from the beginning of the domain area up to the first byte of user code in EEPROM.
	4.401	Field 12	Network-variable alias-table entry size in bytes. Set to 0 if aliases are not supported in the device.
	4.401	Field 13	The base clock-rate factor. Shall be set to either 10000000 or 13107200. This value combined with the protocol-processor clock rate (field 2) determines the base clock rate of the device. If the base clock-rate factor is 10000000, the device clock rate is equivalent to the protocol-processor clock rate defined in field 2. If the base clock-rate factor is 13107200, the device clock rate is equivalent to the protocol-processor clock rate defined in field 2, multiplied by a factor of 1,31072. For example, if field 4 is set to use value 4 (5 MHz), and the base clock-rate factor is set to 13107200, the actual device clock rate is 6,5536 MHz.
Line 8	4.401	Describes the channel parameters. Contains the following fields:	
		Field 1	Boolean that specifies whether a standard transceiver type is used. Set to 1 if a standard transceiver type is used, otherwise set to 0.
		Field 2	Standard Transceiver-Type Identifier. The ID values are listed in Annex C.
		Field 3	Reserved. Set to 1.
		Field 4	Transceiver-interface type. Encoded as follows:
		Value	Type
		0	Not specified
		1	Single-ended mode

Line	Version	Contents		
			2	Special-purpose mode
			5	Differential mode
		Field 5	Transceiver-interface rate. Encoded as follows:	
			Value	Rate
			0	1,25 Mbps
			1	625 kbps
			2	312,5 kbps
			3	156,3 kbps
			4	78,1 kbps
			5	39,1 kbps
			6	19,5 kbps
			7	9,8 kbps
			8	4,9 kbps
			9	2,4 kbps
			10	1,2 kbps
			11	0,6 kbps
		Field 6	Number of priority slots on the channel (0 – 127).	
		Field 7	Minimum clock rate for the channel. Encoded with the same values as the clock rate in line 7 field 2.	
		Field 8	Average packet size in bytes.	
		Field 9	MAC-layer protocol-processor oscillator accuracy in parts per million.	
		Field 10	MAC-layer protocol-processor oscillator wakeup time in microseconds.	
Line 9	4.401	Describes the transceiver parameters. Contains the following fields:		
		Field 1	Channel bit rate in bits per second.	
		Field 2	Special-purpose-mode alternate-channel bit rate in bits per second. Set to 0 for devices that do not use special-purpose-mode transceivers.	
		Field 3	Boolean that specifies whether a special-purpose-mode transceiver controls the preamble. Set to 1 if the transceiver controls the preamble, otherwise set to 0. Set to 0 for devices that do not use special-purpose-mode transceivers.	

Line	Version	Contents
		Field 4 Special-purpose-mode wakeup pin direction. Set to 0 for input, 1 for output. Set to 0 for devices that do not use special-purpose-mode transceivers.
		Field 5 Boolean that specifies whether the device can override the general-purpose data used for special-purpose mode. Set to 1 if the device can override; otherwise set to 0. Set to 0 for devices that do not use special-purpose-mode transceivers.
		Fields 6 – 12 General-purpose data used for special-purpose mode. Set to 0 for devices that do not use special-purpose-mode transceivers.
Line 10	4.401	Describes the channel-timing parameters. Contains the following fields. All field values are in tenths of a bit time, except as noted.
		Field 1 Receive start delay.
		Field 2 Receive end delay.
		Field 3 Indeterminate time.
		Field 4 Minimum interpacket time.
		Field 5 Preamble length.
		Field 6 Turnaround time (microseconds).
		Field 7 Missed preamble time.
		Field 8 Packet qualification time.
		Field 9 Boolean that specifies whether raw data overrides the timing values. Set to 1 if raw data overrides; 0 otherwise.
		Field 10 Raw-data clock rate. Encoded with the same values as the clock rate in line 7 field 2.
		Fields 11 – 15 Raw-data bytes for the communications parameters.
Line 11	4.401	Contains a single asterisk, indicating the end of the transceiver parameters.
Lines 12 – N	4.401	Device self-documentation string (DSDS). If the documentation string is not supplied, there is a single line containing a single asterisk. If supplied, each documentation line begins with a double-quote (") character (not part of the documentation string). Multiple lines shall be concatenated without any intervening characters. There is no end double-quote character; instead, the line is terminated by a newline character. The characters of the string shall all be printable ISO/IEC 646 (ASCII) characters (this includes spaces, but not tabs). Trailing spaces are included. The line may be up to 60 characters long, not including the starting double-quote character or the newline character. Any non-printable characters shall be encoded using a hex-character escape sequence of “\xHH”, where H represents a single hexadecimal digit. The values A – F within a hex-character escape sequence shall be specified exclusively with uppercase letters. If the static interface contains functional blocks, the DSDS shall be formatted as described in EN 14908-5.

Line	Version	Contents
Line N+1	4.401	Blank line.

8.2.2.2 Network Variable and Message Tag Definition Section

8.2.2.2.1 General

This subclause consists of zero or more network variable or message tag definitions. The number of network variable definitions that follow shall be the same as the number of static network variable declarations specified in field 4 of line 6 of the header.

8.2.2.2.2 Network Variable Definition

Following is an example of a network variable definition. The lines are numbered for reference in this document; these line numbers are not included in the device interface file.

1: VAR nvo01Value 2 0 0 0

2: 0 1 63 1 0 1 0 1 0 1 0 0 0

3: "@1|2

4: 95 * 2

5: 1 0 0 0 0

6: 1 0 0 1 0

A network variable definition consists of the following lines:

Line	Version	Contents	
Line 1		A line with the following syntax: VAR *name index avgRate maxRate arraySize* The fields are defined as follows:	
	4.401	name	The network-variable name (maximum of 16 characters without whitespace). This name is also called the *programmatic name*. The name shall be unique within the functional block containing the network variable. If the network variable is not a member of a functional block, the name shall be unique for all the network variables that are not members of functional blocks. Development tools may restrict or prevent the generation of duplicate programmatic names.
	4.401	index	The network-variable index specified as a decimal string (0 – 4095).
	4.401	avgRate	The average-rate estimate specified as an encoded decimal string (0 – 250). Encoded as an unsigned decimal *n*, where the rate estimate = 2(*n*/8)-5. Set to 0 if the estimate is not specified.

Line	Version	Contents	
	4.401	maxRate	The maximum-rate estimate specified as an encoded decimal string (0 – 250). Encoded as an unsigned decimal *n*, where the rate estimate = 2(*n*/8)-5. Set to 0 if the estimate is not specified.
	4.401	arraySize	The number of network variables in a network-variable array, or 0 if this network variable is not an array. Each element of a network-variable array is assigned a unique network-variable index number. The network-variable index number for the entry following that for an array shall be equal to the index number of the first element of the array plus the number of elements in the array.
Line 2	4.401	Contains the following fields:	
		Field 1	Specifies whether the device should be taken offline before updating the variable. Set to 0 if the variable can be updated when online or offline, or 1 if it should be updated only when offline.
		Field 2	Shall be set to 1.
		Field 3	Shall be set to 63.
		Field 4	Network variable direction. Set to 0 for an input, 1 for an output.
		Field 5	Default service type to use for connections containing this variable. Set to 0 for acknowledged, 1 for repeated, or 2 for unacknowledged.
		Field 6	Specifies whether the service type can be changed in the field. Set to1 if the type can be changed, 0 if it cannot.
		Field 7	Specifies the authentication default for the network variable. Set to 1 to use authentication for the network variable by default, 0 to not use authentication by default.
		Field 8	Specifies whether the use of authentication can be changed in the field. Set to 1 if the use of authentication can be changed, 0 if it cannot.
		Field 9	Specifies the default use of priority for the network variable. Set to 1 to use priority for the variable by default, 0 to not use priority by default.
		Field 10	Specifies whether the use of priority can be changed in the field. Set to 1 if the use of priority can be changed, 0 if it cannot.
		Field 11	Specifies the polled attribute of the network variable. For an input, set to 0 if the application program does not poll using this variable, 1 if it does. For an output, set to 0 if the network variable sends unsolicited updates, 1 if the network variable need be polled for updates.
		Field 12	Specifies the synchronized attribute of the network variable. Set to 0 if the network variable is not synchronized, 1 if the network variable is synchronized (*i.e.*, all outputs are transmitted and their order is preserved).
		Field 13	Specifies the configuration attribute of the network variable. Set to 0 for a non-configuration class network variable; 1 for a configuration class network variable.

Line	Version	Contents
Lines 3 – N	4.401	This line and the following lines define the network variable's self-documentation. If the variable has no self-documentation, the line contains a single asterisk. If supplied, one or more lines of text appear here; each line begins with a double-quote character and ends with a newline. The characters of the string shall all be printable ISO/IEC 646 (ASCII) characters (this includes spaces, but not tabs). When the lines are concatenated together without the double-quote or newline characters, this forms the self-documentation text. Each line may be up to 60 characters long not including the double-quote or newline. Any non-printable characters shall be encoded using a hex-character escape sequence of "\xHH", where H represents a single hexadecimal digit. The values A – F within a hex-character escape sequence shall be specified exclusively with uppercase letters. If the variable is part of a functional block, the variable's self-documentation string shall be formatted as described in EN 14908-5.
Line N+1	4.401	The first line after the self-documentation provides network variable type information. The line has the following syntax: *snvtIndex * elementCount* The fields are defined as follows: snvtIndex — Specifies the SNVT index (1 to 255) or 0 if this variable is a user-defined network-variable type. See Clause 4 for index list. * — An asterisk character ("*") preceded and followed by space characters. elementCount — Number of elements (1 to 256) in a network-variable structure or union. Set to 1 if the network variable is not a structure or union.
Lines N+2 – M	4.401	Network-variable characteristics. If the network variable is not a structure or union, there is just one line. If the network variable is a structure or union, there is one line for each data element of the structure or union. Each line has the following syntax: *type offset size signedFlag arraySize* The fields are defined as follows: type — Network-variable data type. One of the following values:

Value	Data Type
0	Character
1	8-bit Integer
2	16-bit Integer
3	Bitfield
4	Union
5	Typeless. None of the remaining fields are applicable.

Line	Version	Contents	
		offset	Network-variable bitfield offset (0 to 7). Set to 0 if the network variable is not a bitfield.
		size	Number of bits in a network-variable bitfield (1 to 7), or the number of bytes in a union (1 to 31). Set to 0 if the network variable is not a bitfield or union.
		signedFlag	Set to 0 if the network-variable type is unsigned, 1 if signed. Set to 0 if not applicable.
		arraySize	Set to 0 if the network-variable type is not an array or, if the type is an array, the size of the array (1 to 31 bytes).

8.2.2.2.3 Message Tag Definition

Following is an example of a message tag definition. The lines are numbered for reference in this document; these line numbers are not included in the device interface file.

1: TAG user_tag 0 69 76 0

2: 0 1 63 1 0 1 0 1 0 1 0 0 0

A message-tag definition consists of the following lines:

Line	Version	Contents	
Line 1		A line with the following syntax:	
		TAG *name index avgRate maxRate zero*	
		The fields are defined as follows:	
	4.401	name	The message-tag name (maximum of 16 characters without whitespace).
	4.401	index	The message-tag index specified as a decimal string (0 – 14).
	4.401	avgRate	The average-rate estimate specified as an encoded decimal string (0 – 250). Encoded as an unsigned decimal *n*, where the rate estimate = 2(*n*/8)-5. Set to 0 if the estimate is not specified.
	4.401	maxRate	The maximum-rate estimate specified as an encoded decimal string (0 – 250). Encoded as an unsigned decimal *n*, where the rate estimate = 2(*n*/8)-5. Set to 0 if the estimate is not specified.
	4.401	zero	Set to 0.
Line 2	4.401	A line with the following syntax:	
		0 *bindFlag* 63 1 0 1 0 1 0 1 0 0 0	
		The bindFlag field specifies whether the tag is bindable. Set to 1 if it is; 0 if it is not. In general, this should be set to 1.	

8.2.2.3 File Definition Section

This section defines the configuration files used for defining configuration properties implemented within configuration files. These files consist of zero or one template file definitions followed by zero, one, or two value file definitions. If a template file is defined, one or two value files shall be defined; however, the contents of value files may be empty.

A file definition consists of the following lines:

Line	Version	Contents
Line 1	4.401	A line with the following syntax: FILE name index type [length] The fields are defined as follows: name The file name (maximum of 16 characters without whitespace). index The file index as defined in the CNP file-transfer protocol in 8.2.2.4. Set to 0 for the template file, or 1 or 2 for the value files. type The file type as defined in the CNP file-transfer protocol in 8.2.2.4. Set to 2 for the template file, or 1 for the value file. length The number of bytes in the file. This value is optional and is calculated from the contents of the file, but shall be specified if the contents of the file are not specified. When not required, the value may be omitted or set to 0. If both the length and file contents are specified, the length value shall equal the number of bytes in the file contents.

Lines 2 – N	4.401	File contents. A line can be interpreted as characters or as binary data.
		
		Character format is indicated by a double-quote (") character as the first non-whitespace character. The double-quote character is not included in the file. In this format, a subsequent double-quote character is considered to terminate the string and it, and all subsequent characters, are not included. Any non-printable characters shall be encoded using a hex-character escape sequence of “\xHH”, where H represents a single hexadecimal digit. The values A – F within a hex-character escape sequence may be specified with either uppercase or lowercase letters. For example, to include a 0x8A character, enter \x8a or \x8A in the string.
		
		Binary format is assumed for any line not starting with a double-quote (") character (excluding whitespace). In binary format, numbers are entered using hex characters. Each value may optionally start with a "0x" or "\x" prefix. Values may optionally be separated with commas or spaces. If separators are not used, every pair of values represents one hex byte. Non-hex-value characters are ignored. For example, each of the following lines generates an identical, four-byte value of 0x0789abcd:
		
		`0x07, 0x89, 0xAB, 0xCD`
		
		`0x0789abcd`
		
		`0789abcd`
		
		`7,89,ab,cd`

Line	Version	Contents
		`\x07\x89\xab\xcd`
Lines N+1	4.401	Blank line.

8.2.2.4 Network Variable Values Definition Section

This subclause defines default values for configuration properties implemented as configuration network variables.

The network variable values definition section consists of the following lines:

Line	Version	Contents
Line 1	4.401	The NVVAL keyword.
Lines 2 – N	4.401	A definition line for each configuration network variable defined in the device interface file. The order of the definitions shall match the order of declaration of the configuration network variables in the device interface file, and there can be no more values than there are configuration network variables in the device interface file. Each line contains the default values, in hex. Each value may optionally start with a "0x" or "\x" prefix. Values may optionally be separated with commas or spaces. If separators are not used, every pair of values represents one hex byte. Non-hex value characters are ignored. For example, the following generates a two-byte value of 0x0789: 0x07, 0x89 0x0789 789 7,89 \x07\x89
Line N	4.401	Blank line.
Line 1	4.401	The NVVAL keyword.

Following is an example network variable values definition. Comments (lines starting with "#") are used to identify each of the values.

```
NVVAL
# unsigned long = 5000
0x13, 0x88
# signed short = 100
0x64
# unsigned long = 2252
0x08, 0xCC
```

9 Standard method of file transfer between devices

9.1 Introduction

The purpose of this subclause is to establish a standard method of exchanging large amounts of data between devices in a CNP network. The largest practical amount of data that can be transferred in a single CNP packet is 228 bytes, but this file-transfer protocol breaks-up data files into packets containing 32 bytes of data and transfers the packets sequentially. The size of the packet is fixed at 32 bytes for interoperability and low device complexity, but may be increased if there is no interoperability requirement. Larger packets may require off-chip RAM to store application and network buffers on a CNP device and thus increase the complexity of the device. This file-transfer method can be used by any CNP device.

Data exchange between devices in the network is modelled as a file transfer, where a file is a stream of bytes, accessed sequentially either for read or for write. A file system is not required, and no assumptions are made about the file system such as format of a filename or existence of any particular directory structure, or file typing. In this way, any device may participate in a file transfer.

Three logical devices participate in a file transfer operation: the *initiator*, the *sender*, and one or more *receivers*. The initiator device may also be the sender or one of the receivers. In this way, a device may initiate the transfer of a file from another device to itself, or a device may initiate the transfer of a file from itself to one or more other devices.

The set-up of a file transfer is implemented with standard network variables that allow the initiator to communicate with the sender and the receivers. The actual transfer itself is implemented with application messages, using a windowed protocol.

9.2 Windowed Transfer Protocol

Most data packets in the file transfer are sent with unacknowledged service, with a request/response packet sent every six packets. These five unacknowledged packets and the sixth request/response packet constitute a *data window*. This windowing protocol avoids the overhead of acknowledging every packet, but allows recovery from a lost packet no more than six packets later. Each packet contains 32 bytes of data, so that standard buffer sizes may be used on each device, without the need for additional RAM space. The response from the receivers to the sender includes an indication of the last packet in the window that was successfully received, incremented by one. The sender needs to buffer the last data window, in case it receives a request from the receivers to retransmit one or more packets from that window. For example, if no errors occur, Figure 1 shows two successive data windows. Packets marked U are unacknowledged. RQ is a request packet from the sender, and RSP is a response from the receiver containing the number of the last packet successfully received incremented by one, which is 5 + 1 = 6.

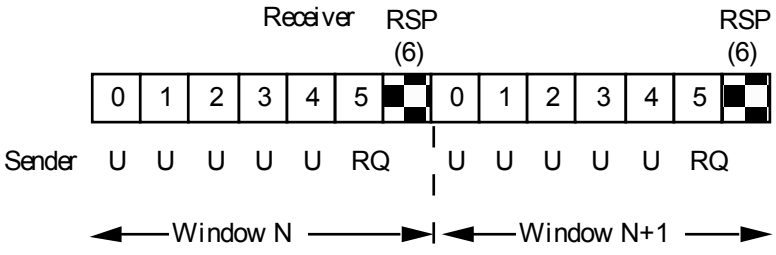

Figure 1 — Window successfully received

Figure 2 shows an example of error recovery. If packet 3 in the window is not received correctly, the response packet RSP contains a 3, and the sender retransmits the window starting with packet 3. Note that packets 3, 4, and 5 are retransmitted to avoid the need for the receiver to buffer packets 4 and 5. In the case of a multicast transfer, the error recovery begins with the lowest numbered packet not received.

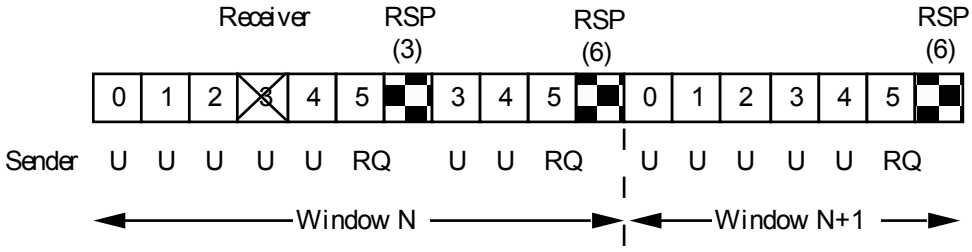

Figure 2 — Window with retransmission

9.3 Setting-Up a File Transfer

Two sets of network variables are used to set-up a file transfer: SNVT_file_req network variables are used for communication from the initiator to the sender and receivers − these are called the *file transfer request network variables*. SNVT_file_status network variables are used for communication from the sender and receivers to the initiator these are called the *file transfer status network variables*. In the case where the initiator is the same as one of the other devices, the network variables are turn-around, meaning that outputs are connected to inputs on the same device. The initiator can choose to monitor and control the file-transfer NVs either by being connected to them or by using explicitly addressed updates and polling.

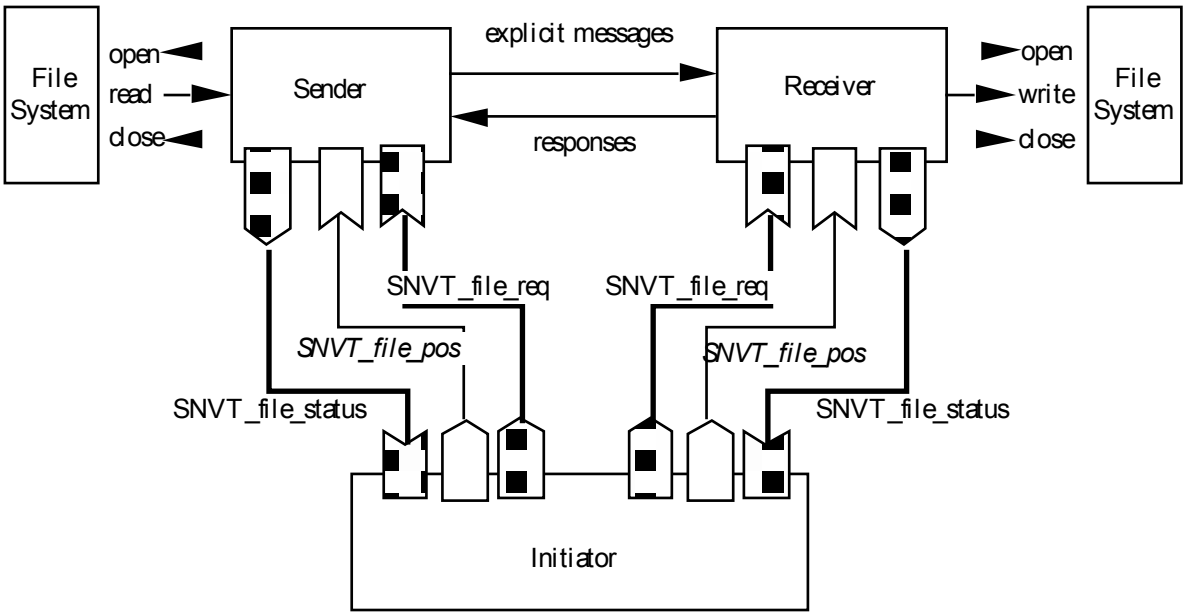

Figure 3 — File Transfer Architecture

To start the transfer, the initiator updates the file transfer request network variable on the receivers in order to open the destination files for writing. The result of the file open operation is returned in an update to the file transfer status network variable on the receivers. If all the receiver files were successfully opened, the initiator then uses the same procedure to open the source file on the sender for reading. At the same time, the initiator passes the network address of the receivers to the sender. The sender uses this address to pass the file data in explicitly addressed messages to the receivers. At the conclusion of the transfer, the sender informs the initiator that it has transmitted the whole file, and the initiator then informs the sender to close the file and receivers to close and save the files.

During data transfer and setup, several potential errors can occur. The sender or receivers can fail to open the specified files. In this case, the initiator backs the other devices out of the operation appropriately. The sender can detect a read error (other than end-of-file), and the receivers can detect a write error. If a receiver fails to receive a data packet within a configurable time-out, it closes and restores its output file, aborts the transfer and informs the initiator. The initiator passes this timeout value to the receivers as part of the setup request.

Similarly, if the sender fails to receive a response to one of its requests to the receivers, it closes its input file, aborts the transfer and informs the initiator. Any error prior to the complete file transfer causes any partially received files to be restored to their original state.

9.4 Random Access

The random access protocol allows a file to be opened and then to be transferred in a piecemeal fashion. The initiator initiates transfers of a subset of the file by performing a seek operation first on the receiver and then on the sender. The seek includes an offset and a length. Multiple seeks may be executed within the scope of a single open/close by the initiator. The length shall be the same in both the sender and receivers while the offsets may differ.

Each seek causes an exchange which is virtually identical to that of a full file transfer. The term data exchange is used to refer to the process which occurs following each seek as well as that during a full file transfer. A full file data exchange differs from that of a seek in that a full file data exchange always starts with window 0 whereas each seek data exchange starts with a window number one greater than the previous (starting with 0).

9.5 Delayed Responses

An initiator normally expects that following the update of the file transfer request NV, the file transfer status NV will immediately be updated to indicate a success or failure condition. However, it is possible that a device accessing files on a disk or network file system may require some extra time to prepare the file transfer status NV response. To this end, the initiator shall tolerate a "working on it" response to either an open, close, or look-up operation. Read, write and seek operations shall be responded to immediately. This may require buffering of data in the device to satisfy this requirement.

The "working on it" response to an open or look-up operation is FS_XFER_OK. For a look-up operation, the file transfer status NV union shall contain the requester address rather than directory information data. The "working on it" response to a close request is either FS_XFER_UNDERWAY or FS_SEEK_WAIT depending on whether a random access transfer is under way.

In a polling situation, upon getting a “working on it” response, the initiator should wait for a period of time of at least one second, before trying again to get the status.

9.6 Completing a Data Exchange

Senders define normal completion of a data exchange as the transmittal of all data within the file or, for random access, transmittal of the number of bytes specified in the seek operation. Receivers define normal completion of a data exchange as the receipt of a file transfer packet with less than the maximum amount of data. It is implementation dependent whether a receiver treats receipt of more or less data than the file length (or seek length for random access) as an error. However, it is recommended that an FS_IO_ERR be returned if the data exchange is too long. The sender shall always transmit the last packet for a data exchange with a length less than the maximum, even if it means transmitting a packet with zero data.

9.7 Completing a File Transfer

Whether there are errors or not, it is always the initiator that closes the files on the sender and the receivers. This guards against race conditions, which could otherwise occur in multiple-initiator scenarios. To guard against the initiator never closing the file, the sender and the receivers shall have the ability to do a local close in the event that the initiator does not close the file in a timely manner, for example, within 1 min after an error condition or normal completion. Furthermore, the initiator shall close the file on the sender first then the receivers. This protects against race conditions where the sender is still sending while the receiver is closed and reopened by another initiator.

9.8 Multicast File Transfers

A multicast file transfer requires that the sender be given a group number as the destination. The group size specified by the initiator should include the sender even if the sender is not a member of the group. The group may also include devices which are neither senders nor receivers as long as there is no chance of ambiguity when they receive the file transfer messages. That is, these devices cannot also be potential receivers. Such devices are not included in the group size and thus do not respond. The typical case of this is where the initiator is connected to the receivers via a group and then tells the sender to use that same group for the transfer.

9.9 Concurrency

It is possible for a single initiator or multiple initiators to concurrently conduct file transfers involving the same sender and/or receivers. This requires that the sender and/or receivers have multiple sets of file transfer NVs. It is the initiator's job to find an available set of file transfer NVs. If multiple file transfer NVs are defined, they shall be defined in NV arrays. A set of file transfer NVs is thus grouped based on their common array index. An exception to this is that CNP devices may delineate a set of NVs by virtue of their belonging to a common functional block.

Regardless of the number of file transfer NVs, a receiver cannot have multiple incoming files in progress because it has no way to differentiate incoming file transfer messages. An exception to this rule is if the receiver and the initiator are the same device then there may be multiple, simultaneous incoming transfers.

Multiple concurrent file transfers using a single set of NVs is possible but problematic. First, it does not work for random access transfer because SNVT_file_pos does not include a file index. Second, it requires that the initiators be connected to the sender/receivers but there is no means for an initiator to determine that such connections are required. Third, it complicates multicast transfers in term of determining which group to use as the sender's destination. For these reasons, this form of operation is not considered to be interoperable.

9.10 SNVT_file_req Data Structure

In order to avoid operating system dependencies, a file on a sender or receiver is identified with a unique 16-bit number called the *file index*. This allows up to 65 535 files to be identified on any device. The file index is used as an argument to file open and directory lookup operations.

The file transfer request network variables contain an operation code and a file index. When a device receives a file transfer request network variable update, it performs the indicated operation, and returns the status of that operation in a file transfer status network variable. The request operation codes are defined as follows:

```
0 FR_OPEN_TO_SEND
1 FR_OPEN_TO_RECEIVE
2 FR_CLOSE_FILE
3 FR_CLOSE_DELETE_FILE
4 FR_DIRECTORY_LOOKUP
5 FR_OPEN_TO_SEND_RA
6 FR_OPEN_TO_RECEIVE_RA
```

The request functions are:

```
FR_OPEN_TO_RECEIVE
```

Opens the indicated file for writing. The receiver executes a file open operation. The request also contains a timeout value in milliseconds to be used to recover from sender failures. The timeout specified shall take into consideration the time needed to obtain the data with a read operation on the sender, and to dispose of the data with a write operation on the receiver. Status returned may be FS_OPEN_FAIL or FS_XFER_UNDERWAY. FS_XFER_OK can also be returned to indicate a delayed response (see "Delayed Responses" above). Once a file is open, the device will reject all further attempts to open a file until the file is closed.

FR_OPEN_TO_SEND

Opens the indicated file for sequential reading. Additional parameters to this request are a destination explicit address, and two booleans to indicate whether authenticated and/or priority messaging should be used. If there is only one receiver, the destination explicit address is a subnet/node address. If there is more than one receiver, it is a group address. The explicit address also contains a retry count and a transaction timer to be used for the request/response message at the end of every window. The sender executes a file open operation, and begins a file transfer by sending packets to the indicated devices on the domain in which the FR_OPEN_TO_SEND was received. Status returned may be FS_OPEN_FAIL or FS_XFER_UNDERWAY. FS_XFER_OK can also be returned to indicate a delayed response (see "Delayed Responses" above). Once a file is open, the device will reject all further attempts to open a file until the file is closed.

FR_CLOSE_FILE

Closes and saves the specified file. Status returned is FS_XFER_OK. The status can also be left at its current value to indicate a delayed response (see Delayed Responses above).

FR_CLOSE_DELETE_FILE

Closes and backs out any changes to the specified file. Status returned is FS_XFER_OK. This is used for backing out of an aborted transfer. Any transferred changes are deleted and the file is restored to the state it was in prior to the start of transfer.

FR_DIRECTORY_LOOKUP

Retrieves directory information for the specified file. Status returned is FS_LOOKUP_OK or FS_LOOKUP_ERR. FS_XFER_OK can be returned to indicate a delayed response (see "Delayed Responses" above).

FR_OPEN_TO_SEND_RA

Same as FR_OPEN_TO_SEND except it opens the indicated file for reading using random access. The normal status is FS_SEEK_WAIT rather than FS_XFER_UNDERWAY.

FR_OPEN_TO_RECEIVE_RA

Same as FR_OPEN_TO_RECEIVE except it opens the indicated file for writing using random access. The normal status is FS_SEEK_WAIT rather than FS_XFER_UNDERWAY.

9.11 SNVT_file_status Data Structure

The status field in the file transfer status structure contains the status of the last honored request to that device. As long as the device is the process of a data exchange, the status is FS_XFER_UNDERWAY. If the device is awaiting a seek operation, the status is FS_SEEK_WAIT. At the end of the transfer, the status becomes FS_XFER_OK. If a file read or write operation fails, the status is FS_IO_ERR. If the transfer was aborted due to a time-out or transaction failure, the status is FS_TIMEOUT_ERR. If a window is received out of sequence, the receiver's status is FS_WINDOW_ERR. The returned status codes are defined as follows:

```
0 FS_XFER_OK
1 FS_LOOKUP_OK
2 FS_OPEN_FAIL
3 FS_LOOKUP_ERR
4 FS_XFER_UNDERWAY
5 FS_IO_ERR
6 FS_TIMEOUT_ERR
7 FS_WINDOW_ERR
8 FS_AUTH_ERR
9 FS_ACCESS_UNAVAIL
10 FS_SEEK_INVALID
```

```
11 FS_SEEK_WAIT
```

The file transfer status structure always contains the number of files on the device, and the index of the file that was the subject of the last operation.

If the last operation was an FR_OPEN_TO_SEND, FR_OPEN_TO_RECEIVE, FR_OPEN_TO_SEND_RA or FR_OPEN_TO_RECEIVE_RA operation, the data structure returned from a sender or receiver to the initiator always contains the full (domain, subnet, node) address of the initiator. This is for the case of multiple initiators when there may be several operations attempted concurrently on the same set of file transfer NVs. Each initiator is responsible for checking its own address against the value returned in the file status structure to ensure that it was granted the requested access. An initiator shall not close a file (i.e. as part of its error handling) unless it was granted access.

If the last operation was a successful FR_DIRECTORY_LOOKUP operation, the status structure contains the directory entry for the specified file. The directory entry is composed of a 16-bit file type, a 32-bit file size, and a 16-character file information array. The latter can be used for any purpose though a nul-terminated ISO/IEC 646 (ASCII) string is recommended. Neither the type nor information string fields have any significance to the file transfer software, they are provided as a convenience to the application. For example, the information string may be used as a file name for a host operating system.

User file types shall be at least 256 to avoid conflicting with file types that may be assigned in future versions of this document.

9.12 SNVT_file_pos Data Structure

This structure is used when the file is opened for random access. It contains a 32-bit file position value, representing a byte offset from the beginning of the file. If the specified offset is beyond the end of file, a status of FS_SEEK_INVALID is returned. This structure also contains a 16-bit byte count which specifies the length of the next file transfer. The file shall be opened using one of the random access modes (FR_OPEN_TO_SEND_RA or FR_OPEN_TO_RECEIVE_RA). If the device does not support random access then it will not have a SNVT_file_pos NV and it returns a status of FS_ACCESS_UNAVAIL to a random access open request.

9.13 Application Protocol Data Unit Structure

Each packet of data is transmitted in an application message with a message code of 62, a one-byte header, and 32 bytes of data (or less for the last packet). The message code of 62 is reserved for the file-transfer protocol as described in this document. The most-significant nibble of the header contains the window number (modulo 16), and the least significant nibble contains the packet number within the window (0-5). The receivers check that packets are received in sequential order, and respond with a request for retransmission starting from the first packet not received correctly within the current window. Receivers also check that windows are received in sequential order. This is to handle the case where one Receiver has received all of a window successfully, and another receiver has not received the first packet of that window. The window number allows the receivers to distinguish a retransmission of the current window from the start of a new window.

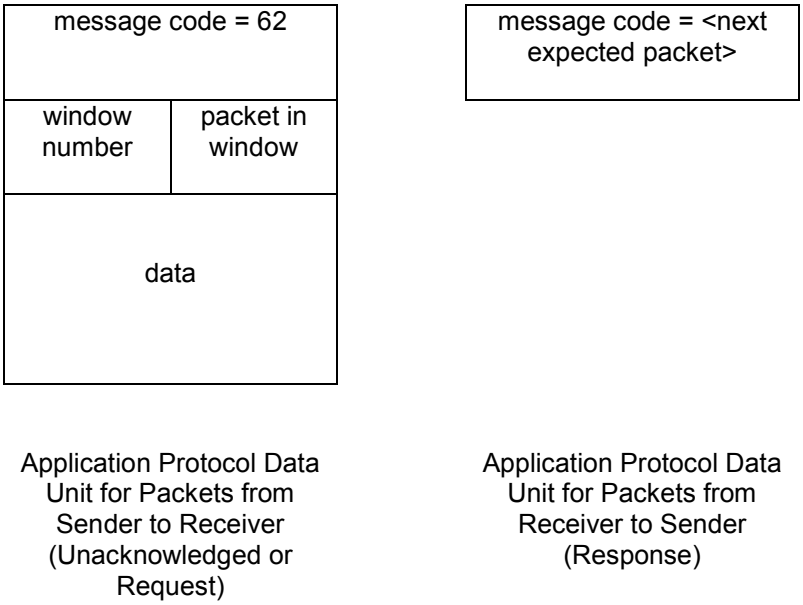

Figure 4 — Application Protocol Data Unit structures

Annex A
(informative)

Protocol Processor Types

This annex lists the Protocol processor model. As described in 8.2.2, Field 1 of Line 7 will be interpreted as:

Table A.1 — Protocol Processor Types

Value	Model
0	Neuron 3150 Chip or FT 3150 Smart Transceiver
1	PL 3150 Smart Transceiver
2	Neuron CY7C53150L Chip
8	Neuron 3120 Chip, Neuron TMPN3120AM Chip, Neuron TMPN3120B1M Chip, Neuron TMPN3120B1AM Chip, or Neuron MC143120B1 Chip
9	Neuron 3120E1 Chip
10	Neuron 3120E2 Chip
11	Neuron 3120E3 Chip
12	Neuron 3120A20 Chip
13	Neuron 3120E5 Chip
14	Neuron CY3120E4 Chip or FT 3120 Smart Transceiver
15	PL 3120 Smart Transceiver
16	Neuron CY7C53120L8 Chip
17	PL 3170 Smart Transceiver

Annex B
(normative)

Standard Program Identifier (SPID) Master List

B.1 General

This annex lists the Standard Program Identifier (SPID) Master List.

B.2 Manufacturer Field

Each manufacturer identifier (MID) is represented in the standard program identifier (SPID) of a device as five (5) hex-digits that are unique to each EN 14908 device manufacturer.

B.3 Device Class Field

Each device class is represented in the standard program identifier (SPID) of a device as four (4) hexadecimal digits identifying the primary function of the device:

50.00 - Access/Intrusion/Monitoring

50.10 --- Intercom

50.11 --- Alphanumeric LCD/Led Display

50.12 --- PS/2 Interface

50.20 --- Motorized Strike Plate

50.30 --- Reader

50.31 -------- Legic® Reader

50.32 -------- Barcode Reader

50.33 -------- Magcard Reader

50.34 -------- Inductive Reader

50.35 -------- Identifier Sensor

50.40 --- Logic Cylinder

50.50 --- Remote Door Manager

50.51 -------- Entry/Exit

50.60 --- Event Memory

50.70 --- Elevator Access Controller

50.80 --- Arming Station

50.90 --- Telephone Controls

50.91 -------- Modem Controller

50.92 -------- Telephone Directory

170.00 - Automated Food Service

172.00 --- Beverage Control and Monitoring

172.10 -------- Heated-Beverage Control and Monitoring

172.11 ---------- Coffee-Making Control and Monitoring

172.20 -------- Cooled-Beverage Control and Monitoring

20.00 - Energy Management

20.10 --- Load-Shed Module (LSM)

20.11 -------- Load-Shed Module with power sensor

20.12 -------- Load-Shed Module with energy monitoring

20.20 --- Circuit Breaker

20.30 --- Scheduler with Presets

21.00 --- 3-Phase Energy Meter

21.01 -------- 3-Phase Demand Power Meter

21.02 -------- 3-Phase Demand Ammeter

21.03 -------- 3-Phase Power Meter

21.04 -------- 3-Phase Ammeter

21.05 -------- 3-Phase Voltmeter

21.10 --- Utility Data Logger Register

21.50 --- Multi-Phase Energy Meter

21.51 -------- Multi-Phase Demand Power Meter

21.52 -------- Multi-Phase Demand Ammeter

21.53 -------- Multi-Phase Power Meter

21.54 -------- Multi-Phase Ammeter

21.55 -------- Multi-Phase Voltmeter

22.00 --- Automated Meter Reader (AMR)

22.01 -------- Utility Meter

110.00 - Fire & Smoke Devices

110.01 --- Fire Smoke Damper Actuator

110.02 --- Smoke (Intelligent) Fire Initiator

110.03 --- Smoke (Conventional) Fire Initiator

110.04 --- Thermal Fire Initiator

110.05 --- Pull-Station Fire Initiator

110.06 --- Audible Fire Indicator

110.07 --- Visible Fire Indicator

110.08 --- Manual Call Point

110.09 --- Fireman Call Point

110.10 --- Universal Fire Initiator

110.11 --- Universal Fire Indicator

70.00 - Gateways

70.10 --- Telephone Gateway

70.11 -------- Alphanumeric Notifier

70.20 --- Internet Gateway

70.21 -------- Internet Web Server

70.22 -------- Internet Information Server

70.30 --- PLC Interface

71.00 --- BMS Gateway

71.10 --- Fire & Smoke Gateways

71.30 --- Industrial Gateways

72.10 --- Refrigeration Gateways

72.20 --- Energy Management Gateways

72.30 --- Lighting Gateways

72.40 --- Vertical/Conveyor Transportation Gateways

72.50 --- Access/Intrusion/Monitoring Gateways

72.80 --- HVAC Gateways

72.90 --- Transportation Gateways

72.91 -------- Signage/Enunciation/Display Systems

07.00 - Generic Actuators

06.00 - Generic Controllers

08.00 - Generic Human-Machine Interfaces (HMIs)

08.10 --- Remote Control

08.11 -------- IR Remote Control

08.12 -------- RF Remote Control

08.20 --- Panel Interface

08.21 -------- Touch-Screen Panel

08.22 -------- LCD/LED Panel HMI

80.00 - HVAC

80.10 --- VAV Controller

80.11 -------- Dual-Duct VAV Controller

80.20 --- Fan Coil Unit Controller

80.30 --- Roof Top Unit Controller

80.40 --- Chiller

80.50 --- Heat Pump (reserved – not for use)

80.51 -------- Heat Pump with Temperature Control

80.60 --- Thermostat

80.70 --- Chilled Ceiling Controller

80.80 --- Unit Ventilator Controller

80.90 --- Space Comfort Control Command Module

81.00 --- SCC - Template for Controller-based nodes (reserved – not for use)

81.10 --- Damper Actuator

81.11 -------- Modulating Damper Actuator

81.20 --- Pump Controller

81.30 --- Valve Controls

81.31 -------- Valve Positioner (HVAC)

Profile-Specific Usage:

00 -- Programmed Actuator Characteristic

01 -- Programmed Actuator Characteristic with Emergency Mode function

20 -- Calculated Actuator Characteristic

21 -- Calculated Actuator Characteristic with Emergency Mode function

81.35 -------- Linear Valve Actuator

82.00 --- Plant Controller

83.00 --- Burner and Boiler Controllers

83.01 -------- Boiler Controller

84.00 --- Product Controllers

85.00 --- SCC - Generic

85.01 -------- SCC - Fan Coil

85.02 -------- SCC - Variable Air Volume

85.03 -------- SCC - Heat Pump

85.04 -------- SCC - Rooftop Unit

85.05 -------- SCC - Unit Ventilator

85.06 -------- SCC - Chilled Ceiling

85.07 -------- SCC - Radiator

85.08 -------- SCC - Air-Handling Unit

85.09 -------- SCC - Self-Contained Unit

85.10 -------- SCC - (reserved, 85.10)

85.29 -------- SCC - (reserved, 85.29)

86.00 --- Air Handling

86.10 -------- Discharge-Air Controller

86.20 -------- Air Handling Critical-Safety Devices

86.21 ---------- Fume Hood Face-Velocity Controller

86.22 ---------- Room Direct-Pressure Controller

05.00 - I/O

05.01 --- Multi-I/O module

05.03 --- Multi-Output Module

05.10 --- 8-Channel configurable I/O

05.11 --- 8-Channel voltage input

05.12 --- 8-Channel current input

05.13 --- 8-Channel digital input

05.14 --- 4 Digital input, 4 digital output

05.15 --- 8-Channel Digital Output

05.16 --- 8-Channel switch input

05.20 --- Analog Input

05.21 --- Analog Output

05.22 --- Extended Analog Input

05.23 --- Extended Analog Output

05.24 --- Generic Analog Input

05.25 --- Generic Analog Output

05.30 --- 6-Channel configurable I/O

05.31 --- 4 Digital input, 2 digital output

05.32 --- 6 Analog outputs U/I

05.40 --- Digital I/O

05.41 --- 7 Digital input, 4 digital output

05.42 --- Digital Input

05.43 --- Digital output

05.48 --- Digital Input Node

05.50 --- 2-Channel configurable I/O

05.51 --- 2-Channel digital input

05.52 --- 2-Channel digital output

05.53 --- 2 Digital input, 2 digital output

05.54 --- 2 Analog input

05.55 --- 2 Analog output

05.60 --- 4-Channel configurable I/O

05.61 --- 4-Channel digital input

05.62 --- 4-Channel digital output

05.63 --- 4 Analog input

05.64 --- 4 Analog output

05.70 --- 12-Channel configurable I/O

05.71 --- 12 Digital inputs Vdc

05.72 --- 12 Digital outputs Vdc

05.73 --- 12 Digital inputs 100-250 Vac

05.74 --- 12 Digital input 12-48 Vac/dc opto iso

05.75 --- 12 Analog inputs U/I/Pt100

05.80 --- Multi-Channel I/O

05.83 --- Multiple Digital Inputs 100-250 Vac

130.00 - Industrial

131.10 --- Generator Set

131.20 --- Automatic Transfer Switch

132.10 --- Programmable Controller

135.00 --- Filtration Systems

135.10 -------- Gas Filtration

135.11 ---------- Fan-Filter Unit

135.20 -------- Liquid Filtration

135.30 -------- Solid Filtration

138.10 --- Remote Miniature Circuit Breaker

138.11 --- Circuit Breaker - Other

138.20 --- PLC Gateway for specific device (reserved – not for use)

138.30 --- Power Supplies

138.31 -------- Power Supply (less than 5V)

138.32 -------- Power Supply (less than 15V)

138.33 -------- Power Supply (less than 50V)

138.34 -------- Power Supply (less than 100V)

138.35 -------- Power Supply (less than 200V)

138.36 -------- Power Supply (less than 500V)

138.37 -------- Power Supply (less than 1 000V)

30.00 - Lighting

30.10 --- Dimmer 1-10V

30.11 -------- 8-Channel dimmer 1-10V

30.12 -------- 3-Channel dimmer 1-10V

30.20 --- Transistor Dimmer

30.30 --- Thyristor Dimmer

30.40 --- Lamp Actuator

30.41 --- ISI Lamp Actuator

30.50 --- Constant Light Controller

30.71 --- Occupancy controller

32.00 --- Switch

Profile-Specific Usage:

00 -- Discrete Output (ON/OFF)

01 -- Discrete Output (ON/OFF) with Feedback Input

02 -- Discrete Output (ON/OFF) with Setting Control

03 -- Discrete Output (ON/OFF) with Feedback Input and Setting Control

20 -- Variable Output

21 -- Variable Output with Feedback Input

22 -- Variable Output with Setting Control

23 -- Variable Output with Feedback Input and Setting Control

32.01 -------- 4-Switch relay

32.02 -------- Multi-switch/multi-sensor

32.03 -------- Switch/sensor

32.50 --- Scene Panel

32.51 --- Scene Controller

32.52 --- Partition Wall Controller

32.53 --- ISI Keypad

33.00 --- Real-Time Keeper

33.01 -------- Real-Time-Based Scheduler

34.00 --- Lighting Controller

34.01 -------- Lighting Panel Controller

35.00 --- Outdoor Lighting

35.12 -------- Outdoor Luminair Controller

60.00 - Motor Controls

60.10 --- Variable-Speed Motor Drive

61.00 --- Sunblind Motor (reserved – not for use)

61.01 -------- Sunblind Motor Controller (reserved – not for use)

61.10 --- Sunblind Actuator

Profile-Specific Usage:

00 -- Feedback Output

01 -- Control (daisy chain) Output

02 -- Feedback Output and Control (daisy chain) Output

61.11 --- Sunblind Controller

61.12 --- ISI Sunblind Actuator

Profile-Specific Usage:

00 -- Feedback Output

01 -- Control (daisy chain) Output

02 -- Feedback Output and Control (daisy chain) Output

54.10 --- AC Drive (reserved – not for use)

01.00 - Network Infrastructure

01.01 --- EN 14908-1 Router

01.02 --- Network Interfaces

01.03 --- Network Managers

01.04 --- Data Logger

01.05 --- Microprocessor Interfaces

01.10 --- Foreign System Connection Sensor

01.20 --- Scheduler

01.30 --- Diagnostic Tools

01.31 -------- Channel Diagnostics

01.32 ---------- Channel Monitor

01.36 -------- Device Monitor

01.37 -------- Channel Continuity Monitor

01.40 --- Generic Controller

01.60 --- Device Manager

04.00 - Programmables

04.01 --- Programer

04.10 --- Static Programmable

04.50 --- Dynamic Programmable

100.00 - Refrigeration

100.10 --- Defrost Controller

100.11 --- Evaporator Controller

100.12 --- Refrigeration Thermostat Controller

100.13 --- Railheat Controller

180.00 - Semiconductor Fabrication

180.10 --- Semiconductor Sensors

180.11 -------- Semiconductor Sensor-AI

180.12 -------- Semiconductor Sensor-AI-AT

180.13 -------- Semiconductor Sensor-AI-MF

180.14 -------- Semiconductor Sensor-AI-Aux

180.15 -------- Semiconductor Sensor-AI-Counter

180.16 -------- Semiconductor Sensor-AI-LCS

180.17 -------- Semiconductor Sensor-AI-SLS

180.18 -------- Semiconductor Sensor-BI

180.19 -------- Semiconductor Sensor-BI-TH

180.20 -------- Semiconductor Sensor-BI-TH-EP

180.30 --- Semiconductor Actuators

180.31 -------- Semiconductor Actuator-AO

180.32 -------- Semiconductor Actuator-AO-MF

180.33 -------- Semiconductor Actuator-BO

180.34 -------- Semiconductor Actuator-EO

180.50 --- Semiconductor Controllers

180.51 -------- Semiconductor Controller

180.70 --- Semiconductor I/O

180.71 -------- Semiconductor Single-Input Single-Output

180.72 -------- Semiconductor Single-Input Single-Output Setpoint

180.80 --- Semiconductor Assemblies

180.81 -------- Semiconductor Assembly

180.82 -------- Semiconductor Assembly-MFM

180.83 -------- Semiconductor Assembly-MFC

180.84 -------- Semiconductor Assembly-ISPM1

180.85 -------- Semiconductor Assembly-ISPM2

180.86 -------- Semiconductor Assembly-ISPM3

180.87 -------- Semiconductor Assembly-ISPM4

180.88 -------- Semiconductor Assembly-ISPM5

180.89 -------- Semiconductor Assembly-ISPM6

180.90 -------- Semiconductor Assembly-ISPM7

180.91 -------- Semiconductor Assembly-ISPM8

180.92 -------- Semiconductor Assembly-ISPM9

180.93 -------- Semiconductor Assembly-ISPM48

180.94 -------- Semiconductor Assembly-EP1

180.95 -------- Semiconductor Assembly-EP2

180.96 -------- Semiconductor Assembly-EP3

180.97 -------- Semiconductor Assembly-EP4

10.00 - Sensors

10.10 --- Light Sensor

10.15 --- Global Solar Radiation Sensor

10.20 --- Time Sensor

10.30 --- Pressure Sensor

10.40 --- Temperature Sensor

10.41 -------- Temperature & Humidity

10.42 --- Frost Sensor

10.50 --- Relative Humidity

10.51 --- Rain Sensor

10.60 --- Occupancy Sensor

10.61 --- ISI Occupancy Sensor

10.70 --- CO2 Sensor

10.80 --- Gas Concentration Sensor

10.81 --- PH Sensor

10.82 --- Conductivity Sensor

10.83 --- Air-Velocity Sensor

10.90 --- Multi-Function Sensor

90.00 - Transportation

90.10 --- Door Controls

91.00 --- Audio Systems

91.10 -------- Railway Audio Systems

91.11 ---------- Railcar Audio Controller

91.12 ---------- Railcar Audio Sensor

95.00 --- Electric Vehicles

95.10 -------- Billing Systems

95.20 -------- Electric Vehicle Supply Equipment

95.21 ---------- EVSE Single, Domestic

95.22 ---------- EVSE Multiple, Domestic

95.25 ---------- EVSE, Commercial

95.30 ---------- EVSE, Retail

140.00 - Vertical/Conveyer Transportation (Elevator)

140.10 --- Elevator/Lift Indicators

140.11 -------- Elevator/Lift Position Indicator and Message Display

140.12 -------- Elevator/Lift Hall Lantern

140.13 -------- Elevator/Lift Arrival Gong

140.14 -------- Elevator/Lift Car-Direction Lantern

140.40 --- Vertical/Conveyer Transportation System Interfaces

140.41 -------- Elevator/Lift Fire-Systems Port

140.60 --- Vertical/Conveyer-Transportation Communication Devices

140.61 -------- Elevator/Lift Voice Announcer

150.00 - Whitegoods

150.10 --- Clothes

150.11 -------- Clothes Washer, Domestic

150.12 -------- Clothes Dryer, Domestic

150.13 -------- Clothes Iron, Domestic

150.14 -------- Clothes Press, Domestic

150.15 -------- Clothes Steamer, Domestic

150.30 --- Cooking

150.31 -------- Oven, Domestic

150.32 -------- Specialty Oven, Domestic

150.33 -------- Microwave or IR Oven, Domestic

150.34 -------- Electric or Inductive Range, Domestic

150.35 -------- Gas Range, Domestic

150.50 --- Storage

150.51 -------- Refrigerator, Domestic

150.52 -------- Deep Freezer, Domestic

150.53 -------- Beverage Cabinet, Domestic

150.54 -------- Dry Cabinet, Domestic

150.70 --- Miscellaneous

150.71 -------- Dishwasher, Domestic

150.72 -------- In-Sink Disposal, Domestic

150.73 -------- Trash Compactor, Domestic

40.00 - Wiring Devices

40.10 --- 1-Pushbutton

40.11 --- 2-Pushbutton

40.12 --- 3-Pushbutton

40.13 --- 4-Pushbutton

40.50 --- Rotary control

41.00 --- AC Load Relay

42.00 --- Controlled Receptacle

B.4 Usage Field

B.4.1 General

Each usage field is represented in the standard program identifier (SPID) of a device as eight bits identifying whether the device has a changeable interface (1 bit), whether the usage is defined by the profile (1 bit), and six (6) bits to specify the usage. The most-significant bit specifies whether the device has a changeable interface. The next bit specifies whether the usage ID specifies a standard usage or a functional-profile specific usage.

B.4.2 Usage ID

The standard usage values for the usage ID are as follows:

Commercial	3
Connectivity	1
General	10
Industrial	2
Industrial/Commercial	4
Medical	9
Network Management	0
Residential	5
Residential/Commercial	6
Telecommunications	8
Transportation	7
Utility	11

B.5 Channel Type Field

Each channel type field is represented in the standard program identifier (SPID) of a device as two hexadecimal digits identifying the channel type supported by the CNP device's transceiver:

IP-852	ID:	154	EN 14908-4
PL-20A	ID:	15	EN 14908-3
PL-20C	ID:	16	EN 14908-3

PL-20N	ID:	17	EN 14908-3	
TP/FT-10	ID:	4	EN 14908-2	
TP/RS485-39	ID:	5	TIA/EIA-485	(Informative)
FO-20L	ID:	152	ANSI/CEA-709.4	(Informative)
FO-20S	ID:	24	ANSI/CEA-709.4	(Informative)

Annex C
(informative)

Standard Transceiver-Type Identifiers

This annex lists the transceiver-type IDs for the channels listed in B.5.

IP-852 (Channel Type ID of 154):	154
PL-20A (Channel Type ID of 15):	20
PL-20C (Channel Type ID of 16):	8
PL-20C-LOW (Channel Type ID of 16):	18
PL-20N (Channel Type ID of 17):	9
PL-20N-LOW (Channel Type ID of 17):	19
TP/FT-10 (Channel Type ID of 4):	7
TP/RS485-39 (Channel Type ID of 5):	3
FO-20L (Channel Type ID of 152):	152
FO-20S (Channel Type ID of 24):	24

Bibliography

[1] TIA/EIA-485:2003, Electrical Characteristics of Generators and Receivers for use in Balanced Digital Multipoint Systems, Telecommunications Industry Association / Electronic Industries Alliance (TIA/EIA), 2500 Wilson Boulevard Suite 300, Arlington VA 22201, USA; Phone: +1-703-907-7700; Fax: +1-703-907-7727; Internet http://www.eia.org/

[2] ANSI/CEA-709.4:1999, Fiber-Optic Channel Specification, Consumer Electronics Association (CEA), 1919 South Eads Street, Arlington VA 22202, USA; Phone: +1-703-907-7600; Fax: +1-703-907-7675; Internet http://www.ce.org/

[3] Directive 2002/91/EC of the European Parliament and of the Council of 16 December 2002 on the energy performance of buildings, OJ L 1, 4.1.2003, pp. 65-71, http://eur-lex.europa.eu/Result.do?T1=V1&T2=2002&T3=91&RechType=RECH_naturel&Submit=Search

[4] Council Directive 89/106/EEC of 21 December 1988 on the approximation of laws, regulations and administrative provisions of the Member States relating to construction products, OJ L 40, 11.2.1989, pp. 12-26
http://eur-lex.europa.eu/Result.do?T1=V1&T2=1989&T3=106&RechType=RECH_naturel&Submit=Search

[5] ISO/IEC 646, *Information technology — ISO 7-bit coded character set for information interchange*

This page deliberately left blank

This page deliberately left blank

British Standards Institution (BSI)

BSI is the national body responsible for preparing British Standards and other standards-related publications, information and services.

BSI is incorporated by Royal Charter. British Standards and other standardization products are published by BSI Standards Limited.

About us

We bring together business, industry, government, consumers, innovators and others to shape their combined experience and expertise into standards -based solutions.

The knowledge embodied in our standards has been carefully assembled in a dependable format and refined through our open consultation process. Organizations of all sizes and across all sectors choose standards to help them achieve their goals.

Information on standards

We can provide you with the knowledge that your organization needs to succeed. Find out more about British Standards by visiting our website at bsigroup.com/standards or contacting our Customer Services team or Knowledge Centre.

Buying standards

You can buy and download PDF versions of BSI publications, including British and adopted European and international standards, through our website at bsigroup.com/shop, where hard copies can also be purchased.

If you need international and foreign standards from other Standards Development Organizations, hard copies can be ordered from our Customer Services team.

Subscriptions

Our range of subscription services are designed to make using standards easier for you. For further information on our subscription products go to bsigroup.com/subscriptions.

With **British Standards Online (BSOL)** you'll have instant access to over 55,000 British and adopted European and international standards from your desktop. It's available 24/7 and is refreshed daily so you'll always be up to date.

You can keep in touch with standards developments and receive substantial discounts on the purchase price of standards, both in single copy and subscription format, by becoming a **BSI Subscribing Member**.

PLUS is an updating service exclusive to BSI Subscribing Members. You will automatically receive the latest hard copy of your standards when they're revised or replaced.

To find out more about becoming a BSI Subscribing Member and the benefits of membership, please visit bsigroup.com/shop.

With a **Multi-User Network Licence (MUNL)** you are able to host standards publications on your intranet. Licences can cover as few or as many users as you wish. With updates supplied as soon as they're available, you can be sure your documentation is current. For further information, email bsmusales@bsigroup.com.

BSI Group Headquarters

389 Chiswick High Road London W4 4AL UK

Revisions

Our British Standards and other publications are updated by amendment or revision.

We continually improve the quality of our products and services to benefit your business. If you find an inaccuracy or ambiguity within a British Standard or other BSI publication please inform the Knowledge Centre.

Copyright

Useful Contacts:

Customer Services
Tel: +44 845 086 9001
Email (orders): orders@bsigroup.com
Email (enquiries): cservices@bsigroup.com

Subscriptions
Tel: +44 845 086 9001
Email: subscriptions@bsigroup.com

Knowledge Centre
Tel: +44 20 8996 7004
Email: knowledgecentre@bsigroup.com

Copyright & Licensing
Tel: +44 20 8996 7070
Email: copyright@bsigroup.com

bsi.

...making excellence a habit.™